

**ESCUELA TÉCNICA SUPERIOR DE INGENIEROS  
INDUSTRIALES Y DE TELECOMUNICACIÓN**

**UNIVERSIDAD DE CANTABRIA**



***Trabajo Fin de Grado***

**ESTUDIO Y CARACTERIZACIÓN DE  
CÉLULAS FOTOVOLTAICAS MEDIANTE  
ARDUINO**

**(Study and characterization of photovoltaic  
cells through Arduino)**

**Para acceder al Título de**

**Graduado en  
Ingeniería de Tecnologías de Telecomunicación**

**Autor: Moisés Bravo Santos**

**Junio-2020**

# **GRADUADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN**

CALIFICACIÓN DEL TRABAJO FIN DE GRADO

**Realizado por: Moisés Bravo Santos**

**Director del TFG: Jesús Mirapeix Serrano**

**Título:** “ Estudio y caracterización de células fotovoltaicas mediante Arduino”

**Title:** “ Study and characterization of photovoltaic cells through Arduino “

**Presentado a examen el día: 29 de junio de 2020**

para acceder al Título de

## **GRADUADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN**

Composición del Tribunal:

Presidente (Apellidos, Nombre): Adolfo Cobo García

Secretario (Apellidos, Nombre): Jesús Mirapeix Serrano

Vocal (Apellidos, Nombre): Pedro Corcuera Miro-Quesada

Este Tribunal ha resuelto otorgar la calificación de: .....

Fdo.: El Presidente

Fdo.: El Secretario

Fdo.: El Vocal

Fdo.: El Director del TFG  
(sólo si es distinto del Secretario)

Vº Bº del Subdirector

Trabajo Fin de Grado N°  
(a asignar por Secretaría)

# Índice

<b>0. Objetivos</b>	<b>4</b>
<b>1. Contexto</b>	<b>5</b>
<b>2. Energía solar FV</b>	<b>6</b>
<b>2.1 Historia Energía solar FV</b>	<b>6</b>
<b>2.2 Contexto y situación global actual</b>	<b>7</b>
<b>2.3 Introducción a la energía solar FV</b>	<b>10</b>
<b>2.4 Efecto Fotoeléctrico</b>	<b>10</b>
<b>2.5 Células fotovoltaica: Tecnologías</b>	<b>12</b>
<b>2.6 Células fotovoltaica: Instalaciones</b>	<b>15</b>
<b>3. Hardware open source</b>	<b>17</b>
<b>3.1 Introducción al hardware open source</b>	<b>17</b>
<b>3.2 Introducción a Arduino</b>	<b>18</b>
<b>3.3 Tipos de Arduino en el Mercado</b>	<b>19</b>
<b>3.4 Entorno de desarrollo de Arduino</b>	<b>22</b>
<b>4. Caracterización de células FV</b>	<b>24</b>
<b>4.1 Característica I-V</b>	<b>24</b>
<b>4.2 Influencia de la irradiancia</b>	<b>28</b>
<b>4.3 Influencia de la temperatura</b>	<b>29</b>
<b>5. Componentes del sistema</b>	<b>31</b>
<b>5.1 Placa Arduino Leonardo</b>	<b>31</b>
<b>5.2 Sensor voltaje FZ0430</b>	<b>32</b>
<b>5.3 Sensor corriente ACS712</b>	<b>34</b>
<b>5.4 Sensor temperatura DHT11</b>	<b>37</b>
<b>5.5 Pulsador de 4 patas para Arduino</b>	<b>39</b>
<b>5.7 Estructura para las mediciones</b>	<b>43</b>
<b>5.8 Tiras LED con fuente de alimentación</b>	<b>48</b>
<b>5.9 Aplicación conexión con Arduino</b>	<b>48</b>
<b>6. Guion de la práctica</b>	<b>49</b>
<b>7. Presupuesto</b>	<b>71</b>
<b>8. Conclusiones y líneas futuras</b>	<b>71</b>

# 0. Objetivos

Este proyecto está enfocado en la realización de una práctica para la asignatura de “Energías y Telecomunicaciones”.

Esta práctica consiste en la caracterización de células fotovoltaicas en un entorno controlado, donde el alumno podrá interactuar con la célula fotovoltaica aplicándole distintos estímulos y observando su respuesta.

Se desarrollará un guion completamente guiado para que el alumno se centre en la caracterización de la célula y en la adquisición de los conceptos básicos asociados.

La principal motivación y objetivo de esta práctica es que los alumnos puedan **aplicar la teoría impartida en clase de una manera práctica y amena**.

Además, y como un paso previo, el alumno deberá revisar un pequeño guion que le servirá para comprender la funcionalidad de los sensores y por qué usamos esos sensores en concreto en nuestra práctica. El alumno deberá leer el guion antes de la realización la práctica, debido a que se le introducirá en la programación de Arduino de una manera muy sencilla. Como parte de las actividades de la práctica se propone que el **alumno complete el esquema de configuración** de los pines de entrada de los sensores y también **el código adyacente a esa configuración** elegida, así el alumno puede adentrarse en el mundo del hardware open-source con Arduino.

Para caracterizar la célula de una manera fiable deberíamos usar un simulador del espectro solar, pero al tratarse de una práctica orientada al hardware open-source y a la caracterización de células fotovoltaicas nos hemos decidido a implementar unas tiras leds en la parte superior de un soporte, para comprobar la diferencia con el espectro solar se obtendrá el espectro y se comparará con el espectro solar.

Otro aspecto importante para los alumnos es ver como alterando las condiciones ambientales tales como la temperatura, el comportamiento de la célula varia. El alumno realizará mediciones a temperatura ambiente y una vez capturadas se subirá la temperatura ambiente unos cuantos grados para comprobar el efecto del calor en la respuesta de las células fotovoltaicas.

Por último, para nuestra herramienta de monitorización de la célula necesitaremos un soporte donde converjan todos los sensores, la fuente de luz y la célula fotovoltaica. Hemos empleado una **impresora 3D** para fabricar un soporte, el cual estará formado por una base, donde se colocará la célula fotovoltaica, sensores y la placa de Arduino. Unas piezas intermedias que su función es la de añadir o disminuir la distancia entre la célula fotovoltaica y la fuente de luz. Por último, la pieza superior donde se sitúa la fuente de luz en forma de tiras leds. Gracias al uso de una impresora 3D los alumnos podrán ver todas las funcionalidades que aporta esta tecnología, siendo su imaginación su única limitación.

# 1. Contexto

Este trabajo se enmarca en la implementación de una práctica para la asignatura G822 (*“Energía y Telecomunicación”*) y G1895 (*“Energy and Telecommunications”*) impartida en el 3º curso del Grado en Ingeniería de Tecnologías de Telecomunicaciones de la Universidad de Cantabria.

Con este trabajo se pretende que el alumnado complete los conceptos teóricos vistos en la asignatura sobre la célula fotovoltaica, en particular sobre su caracterización eléctrica y el efecto de parámetros externo como la temperatura sobre su rendimiento.

Este trabajo abarca la necesaria implantación de una práctica en la que el alumno aborde una herramienta de hardware open-source que, como sabemos, son muy usadas en la actualidad en multitud de aplicaciones y por ello creemos necesario que el alumno se inicie en este mundo de una manera guiada como en la práctica propuesta.

## 2. Energía solar FV

### 2.1 *Historia Energía solar FV*

Cuando hablamos de la historia de la energía solar FV primero tenemos que hacer referencia al descubrimiento del efecto fotoeléctrico. Einstein recibió el premio Nobel de Física por la **explicación del efecto fotoeléctrico** en el año 1921 y no por su teoría de la relatividad. Cabe destacar que Einstein no descubre el efecto fotoeléctrico, sino que fueron otros, años antes los que descubren este fenómeno. En el año 1839 Edmun Becquerel descubre el efecto fotoeléctrico, en 1873 Willoughby Smith descubre el efecto fotoeléctrico en sólidos (selenio) y finalmente en 1877 Adams y Day producen la primera célula fotovoltaica de selenio.

No fue hasta 1954 cuando se produce la primera célula de silicio por los investigadores de los Laboratorios Bell, a partir de este hecho empieza la competencia por conseguir aumentar el **rendimiento de las células fotovoltaicas**. La eficiencia de una célula fotovoltaica es un término muy importante porque representa la eficiencia con la que la placa fotovoltaica convierte la radiación solar en energía eléctrica.

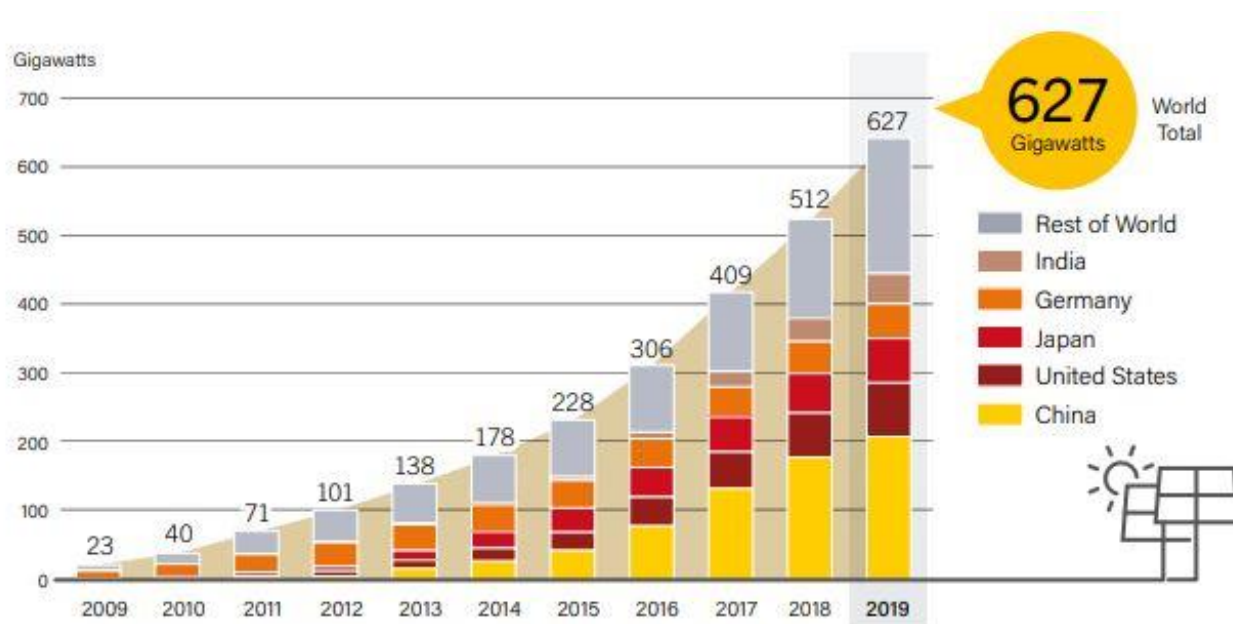
En el año 1957 Hoffman Electronic alcanza un rendimiento del 14% en sus células, en el año 1992 la universidad del sur de Florida alcanza un 16%, en el año 2012 Solar Frontier alcanza un 18%, en el año 2015 Panasonic alcanza un 22,5% y por último en el año 2016 NREL y CSEM alcanzan un 30 %.

Cabe destacar que el desarrollo tecnológico de las células fotovoltaicas estuvo muy ligado con el sector aeroespacial. En el año 1958 se lanza el Vanguard I, el primer satélite alimentado con energía solar fotovoltaica, llevaba 0.1W en una superficie aproximada de  $100\text{ cm}^2$ , para alimentar un transmisor de 5MW que estuvo operativo durante 8 años. En 1962 se lanza el primer satélite comercial de telecomunicaciones, el Telstar, con una potencia fotovoltaica de 14W.

Gracias a todo el desarrollo del sector aeroespacial, la tecnología evolucionó considerablemente y el siguiente gran desarrollo tecnológico se debió a las grandes instalaciones orientadas a la generación eléctrica para su **inyección en la red**. En 1980 ARCO Solar (Shell Solar) es la primera empresa con una producción industrial de 1 MW de módulos al año, para 1983 la producción mundial excede los 20 MW. Durante los siguientes años la potencia instalada mundial siguió creciendo hasta la última década donde el crecimiento es exponencial.

## 2.2 Contexto y situación global actual

En la actualidad la energía solar fotovoltaica juega un papel muy importante dentro de la sociedad y entre los principales actores en instalaciones normalizadas *per cápita* están Alemania, Japón, Italia y Bélgica. Cuando nos referimos a potencia total instalada los líderes son China y USA. En Alemania la potencia instalada es de 43 GW, en Italia 20GW, en Japón 58GW, en china de 205GW, en USA de 82GW y en el mundo la **potencia instalada mundial estimada es de 627 GW**.



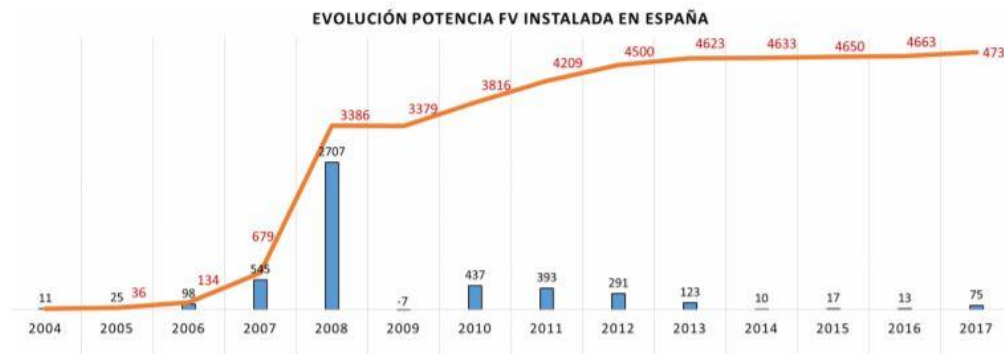
**Figura 1.** Evolución del parque solar fotovoltaico global entre los años 2009 y 2019. Fuente:REN21 (2020 Report). [https://www.ren21.net/wp-content/uploads/2019/05/gsr\\_2020\\_full\\_report\\_en.pdf](https://www.ren21.net/wp-content/uploads/2019/05/gsr_2020_full_report_en.pdf)

El crecimiento mundial de la energía solar fotovoltaica instalada sigue una curva aproximadamente **exponencial** durante la última década, como podemos ver en la Figura 1, donde se ha representado la evolución de la **potencia instalada en los principales países**.

El aumento de los módulos fotovoltaicos instalados está claramente correlacionado con el **coste asociado a los paneles fotovoltaicos**, que como ocurre generalmente con las economías de escala, el aumento en la producción de módulos fotovoltaicos por una determinada empresa ocasiona una **reducción en el coste unitario** de los módulos.

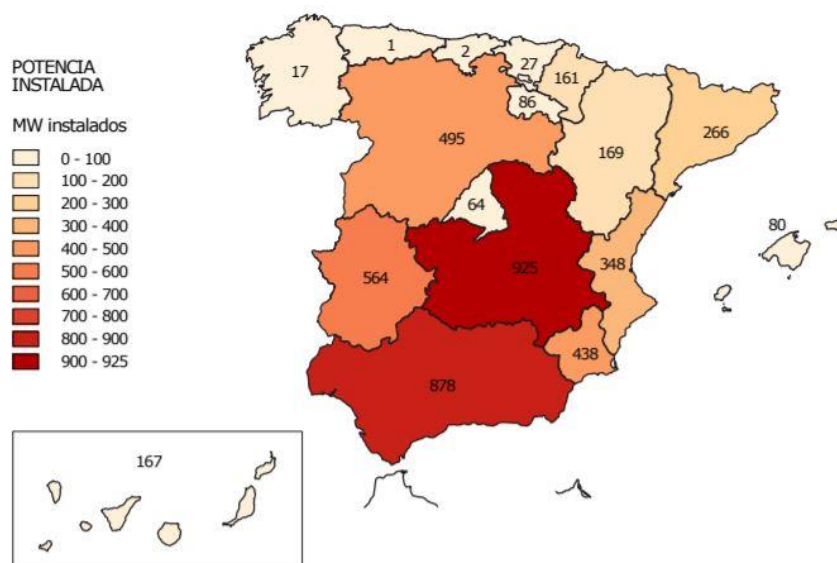
Debido a este descenso en el coste de los módulos fotovoltaicos y junto a la necesidad de cambiar la forma de producir energía eléctrica, ocasionado por el cambio climático y la alta contaminación en las grandes ciudades, es muy importante que aumente el despliegue de la energía solar fotovoltaica durante las siguientes décadas.

En cuanto a la **situación nacional**, el sector de la energía solar fotovoltaica ha sufrido muchas variaciones durante los últimos años y completamente distinta a los países líderes en este ámbito. España alcanzó sus mayores repuntes en el año 2008 debido a las buenas políticas implantadas por el gobierno como **subsidios o primas a las renovables** con esto pretendían fomentar el uso de energías renovables y fue muy exitosa debido a que establecía un precio elevado de compra a esos KWh, lo que implicaba una gran rentabilidad. Pero ese mismo año España sufrió una de las peores **crisis financieras** y la inseguridad generada por los cambios en estas políticas **generaron un estancamiento** en el crecimiento de esta tecnología en nuestro país como podemos ver en la Figura 2.



**Figura 2.** Evolución de la potencia instalada en España hasta el 2017. Fuente: APPA.

La energía solar fotovoltaica es la **tercera tecnología** más importante dentro del mix de generación renovable en España, alrededor de un 10% de generación en los últimos años, siendo superada por la energía eólica e hidráulica. También debemos destacar el lugar de emplazamiento de esta tecnología, factor muy importante debido a que la irradiancia es distinta dentro del territorio nacional. Castilla La Mancha, Andalucía, Extremadura, Castilla León y Murcia suman aproximadamente el 70 % de la capacidad fotovoltaica española total como se puede ver en la Figura 3. Las comunidades del norte de España tienen cifras muy bajas especialmente Galicia, Asturias, **Cantabria** y País Vasco, ya que la generación fotovoltaica tiene una fuerte correlación asociada a la radiación solar que es mucho mayor en las regiones del sur de España.



**Figura 3.** Potencia solar fotovoltaica (MW, finales del 2017) . Fuente: UNEF. <http://bit.ly/2QG0iO5>



A pesar del claro estancamiento del sector fotovoltaico solar en nuestro país en los últimos años, su futuro en los próximos años parece muy prometedor. La potencia fotovoltaica instalada en España creció **4 GW entre 2007 y 2012**, para mostrar un crecimiento anecdótico por los motivos explicados anteriormente, **48 MW entre 2014 y 2017**. Sin embargo, a lo largo **2019** se otorgaron a través de subastas **3.9 GW**, lo que supone un gran impulso para el sector, implicando una inversión de 3500 millones de euros.



#### Potencia instalada nacional (MW)

	2016	2017	2018	2019	2020
Hidráulica convencional y mixta	17.030	17.028	17.046	17.085	17.085
Bombeo puro	3.329	3.329	3.329	3.329	3.329
Nuclear	7.573	7.117	7.117	7.117	7.117
Carbón	10.004	10.004	10.030	9.683	9.456
Fuel + Gas	2.490	2.490	2.490	2.447	2.447
Ciclo combinado	26.670	26.670	26.284	26.284	26.284
Hidroeléctrica	11	11	11	11	11
Resto hidráulica <sup>(1)</sup>	-	-	-	-	-
Eólica	23.001	23.082	23.545	25.799	25.902
Solar fotovoltaica	4.683	4.685	4.712	8.913	9.524
Solar térmica	2.304	2.304	2.304	2.304	2.304
Térmica renovable/Otras renovables <sup>(2)</sup>	870	872	877	1.076	1.076
Térmica no renovable/Cogeneración y resto/Cogeneración <sup>(3)</sup>	5.965	5.801	5.727	5.677	5.672
Residuos no renovables <sup>(4)</sup>	496	496	490	490	490
Residuos renovables <sup>(4)</sup>	160	160	160	160	160
<b>Total</b>	<b>104.588</b>	<b>104.050</b>	<b>104.123</b>	<b>110.376</b>	<b>110.858</b>

**Figura 4.** Potencia instalada en España a principios del 2020. Fuente: REE. <http://bit.ly/37ZRclK>

Los datos proporcionados por REE para 2019 respaldan claramente este cambio, **en 2019 la capacidad fotovoltaica aumento en España en 4200MW**. Todos estos datos son mucho más elocuentes si atendemos a los proyectos que ya han solicitado los permisos REE correspondientes, en particular los permisos de acceso y conexión a la red eléctrica. A finales de 2019 hay solicitudes con el permiso de acceso otorgado por un total de 75.7 GW, con 28.2 GW adicionales con gestión de acceso en progreso y 55.2 GW con la solicitud denegada en este momento. Estos datos confirman el cambio en el sector durante el 2019.

Pero como podemos observar en la Figura 4 durante el primer trimestre de 2020 **la potencia instalada con respecto al 2019 con fecha de mayo de 2020 es de 600 MW** un claro decremento en el sector. Este decremento se debe a la **pandemia global COVID-19** debido a la crisis financiera que esta pandemia está causando en nuestra sociedad. Una vez más el sector de la energía solar fotovoltaica se ve afectada es un momento de gran crecimiento tal y como paso en el año 2008 con la crisis financiera.

## 2.3 *Introducción a la energía solar FV*

La energía solar es una de las energías renovables que más evolución han tenido durante las últimas décadas y seguramente una de las más importantes durante las siguientes décadas futuras debido a que el sol arroja sobre la tierra **cuatro mil veces más energía** de la que se consume anualmente en nuestro planeta y sin necesidad de gastar recursos limitados.

Aunque también debemos tener en cuenta que esta energía **no siempre está disponible**, estamos limitados a unas horas al día y sobre todo a la climatología, por eso es de vital importancia perfeccionar la eficiencia de las células fotovoltaicas y sobre todo el **almacenamiento** de esta energía en baterías para poder disponer de energía en cualquier momento sin depender de energías no renovables.

## 2.4 *Efecto Fotoeléctrico*

El efecto fotoeléctrico es el principio por el cual una célula fotovoltaica convierte la radiación solar en electricidad, esta radiación solar está compuesta por fotones que tienen distinto nivel de energía dependiendo de su longitud de onda. El concepto es que esos fotones se transforman en energía eléctrica, pero cuando la luz incide sobre la célula, los fotones pueden ser reflejados, absorbidos y a veces atravesar la célula sin crear ninguna interacción, solo los **fotones absorbidos** pueden llegar a producir energía eléctrica.

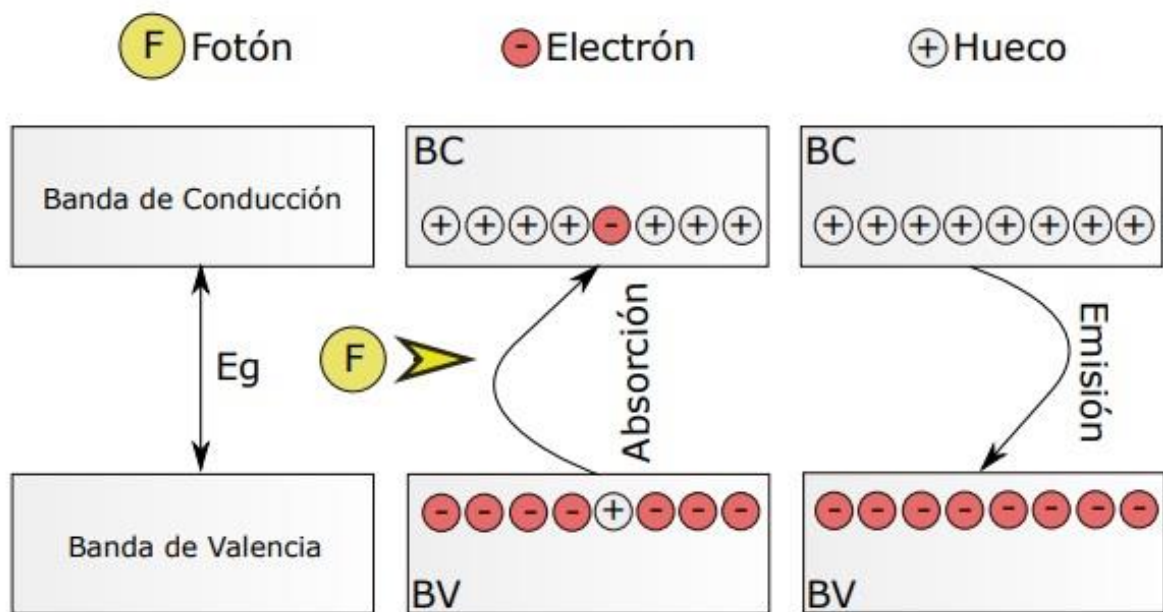
Las células fotovoltaicas se fabrican con materiales semiconductores, como el silicio, estos materiales se comportan como un conductor o como un aislante dependiendo de diversos factores como la temperatura, la radiación o el campo eléctrico. El silicio es un material de gap indirecto, esto quiere decir que la transición de un electrón de la banda de valencia a la de conducción será más complicada que en uno de gap directo. Además, presenta una respuesta espectral que no es óptima en cuanto al máximo aprovechamiento de la radiación solar debido a que la respuesta espectral del silicio llega hasta aproximadamente a **1100nm** y el espectro de la radiación solar llega hasta los **2000 nm**.

Una célula fotovoltaica es básicamente una **unión PN** con una superficie lo suficientemente grande para **capturar el mayor número de fotones posible**. Para formar una unión PN solo hay que emplear una base de silicio y generar una parte p y otra n. La parte p se genera incluyendo un elemento dopante con 3 electrones en la capa de valencia, dando lugar a un enlace incompleto, creando **huecos**. Si se añaden suficientes átomos de tipo p, los huecos serán los portadores mayoritarios. De la misma manera se puede generar una parte n, donde los electrones serán ahora los portadores mayoritarios al dopar con un elemento como el fósforo que tiene 5 electrones en la capa de valencia. Si las partes p y n se unen con un material conductor, los mayoritarios de cada parte tenderán a desplazarse a la zona opuesta. Cuando un hueco y un electrón se encuentran en la zona de unión, ambos se neutralizan y desaparecen provocando que la zona n pierda electrones y la zona p pierda huecos generando una diferencia de potencial.

Cuando la radiación solar incide sobre la célula fotovoltaica una parte de esta radiación es absorbida dentro del material, cuando esos fotones son absorbidos por un átomo su energía se emplea para subir a un electrón de la banda de valencia a la de conducción.

Para ello, la energía del fotón ha de ser al menos igual a la energía del gap del material como podemos ver en la Figura 5.

Como los electrones tienden a concentrarse en la célula fotovoltaica donde incide la radiación solar, se genera un campo eléctrico con dos zonas bien diferenciadas: la negativa, en la cara iluminada donde están los electrones y la positiva en la zona opuesta donde están los huecos. Si ambas zonas se conectan a través de conductores se produce un desequilibrio eléctrico originando una diferencia de potencial mencionada anteriormente, creando una corriente eléctrica continua para igualar las cargas.



**Figura 5.** Ilustración de los procesos de absorción y emisión en un semiconductor. Fuente: Apuntes de la asignatura “Energía y Telecomunicaciones”.

No todos los fotones que nos llegan del Sol pueden ser empleados para generar corriente eléctrica debido a que como hemos comentado antes la **respuesta espectral del silicio está limitada a los 1100nm**.

Además, si la energía del fotón incidente es menor que la energía del gap del material no existirá interacción y el fotón atravesará la célula fotovoltaica. Si estas energías son iguales el fotón es absorbido y liberará un electrón que contribuirá a la generación de corriente eléctrica. Aunque si la energía del fotón es mayor que la energía del gap del material, el fotón será absorbido pero el exceso de energía generará un efecto térmico negativo que puede comprometer el rendimiento de la célula.

Por último, el concepto **eficiencia** de una célula es uno de los más importantes debido a su relación entre los fotones que inciden y la corriente generada. Se define como el *ratio* de conversión de la energía proveniente del Sol en energía eléctrica y cuya formula es la siguiente:

$$Eficiencia = \frac{PotenciaFV(electrica)(W)}{Irradiancia(\frac{W}{m^2}) \cdot Superficie(m^2)}$$

Hoy en día, en paneles fotovoltaicos comerciales, podemos hablar de **eficiencias del orden de 14% a algo superior al 20%** para tecnologías de silicio cristiano.

En **entornos experimentales** se puede conseguir **hasta el 60%** pero con procedimientos complejos y costosos que hacen muy difícil su comercialización.

## 2.5 Células fotovoltaica: Tecnologías

Existen diversas tecnologías de células fotovoltaicas dependiendo del **material** empleado para su fabricación. Podemos hacer una división entre las distintas tecnologías, unas basadas en obleas de **materiales semiconductores** como el silicio y otras basadas en tecnologías de **capa fina**.

En la **primera generación** de células fotovoltaicas predomina el **silicio cristalino** que representa el **90%** del mercado mundial de células fotovoltaicas, incluyendo las tecnologías de silicio **monocristalino y policristalino** que tienen gran superficie, alta calidad y se pueden unir fácilmente. Estas tecnologías no permiten ya avances significativos en la reducción de los costes asociados a su producción.

El **silicio monocristalino** es de una calidad superior al policristalino lo que conlleva un precio superior. Cuando hablamos de eficiencia el silicio monocristalino presenta unos valores típicos del 18% mientras que el policristalino se sitúa alrededor del 14%.

Una forma de diferenciar una célula o panel fotovoltaico monocristalino y policristalino es su aspecto visual, ya que el monocristalino presentar un **color negro uniforme** mientras que el policristalino presenta un **color azulado y no uniforme** en su superficie como podemos ver en la Figura 6.

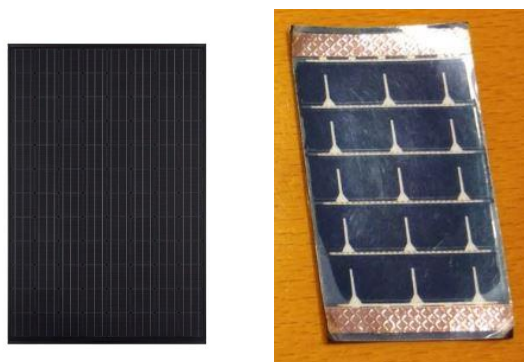


**Figura 6.** Comparación de una célula monocristalina (izquierda) con una policristalina (derecha).

La **segunda generación** de células fotovoltaicas se corresponde con las células elaboradas a partir de **compuestos químicos**, algunos de estos materiales son las películas finas de telurio de cadmio (CdTe), de silicio amorfo y de silicio microamorfo.

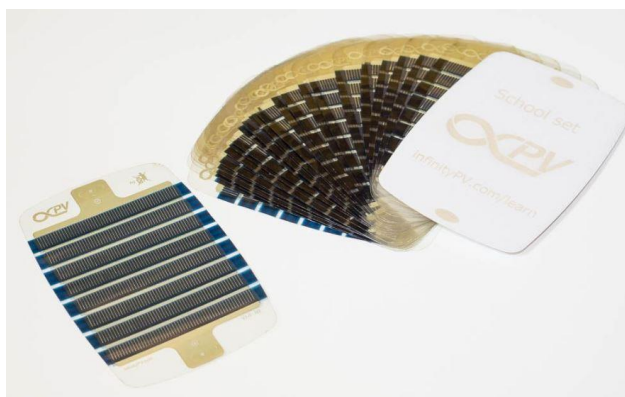
Dentro del conjunto de células de la segunda generación tenemos las células de **capa fina** que presentan un espesor inferior a las de primera generación alrededor de los 5µm lo que provoca que el peso sea menor que las de silicio cristalino pero una eficiencia menor alrededor de 6-8%. Otra ventaja es que la tecnología de capa fina permite que se fabriquen células con **más de una capa** de diferentes materiales semiconductores, aumentando así la eficiencia del panel fotovoltaico al combinar las respuestas espectrales de dichos materiales.

Dentro de las tecnologías de capa fina podemos encontrar las de **silicio amorfo**, su penetración en el mercado se reduce a un 2% aproximadamente. La fabricación es completamente distinta a las de silicio cristalino ya que en este caso se trabaja con el **silicio en estado gaseoso** y resulta mucho más sencillo su fabricación que las de silicio cristalino.



**Figura 7.** Célula fotovoltaica de silicio amorfo(izquierda) y de capa fina (derecha).

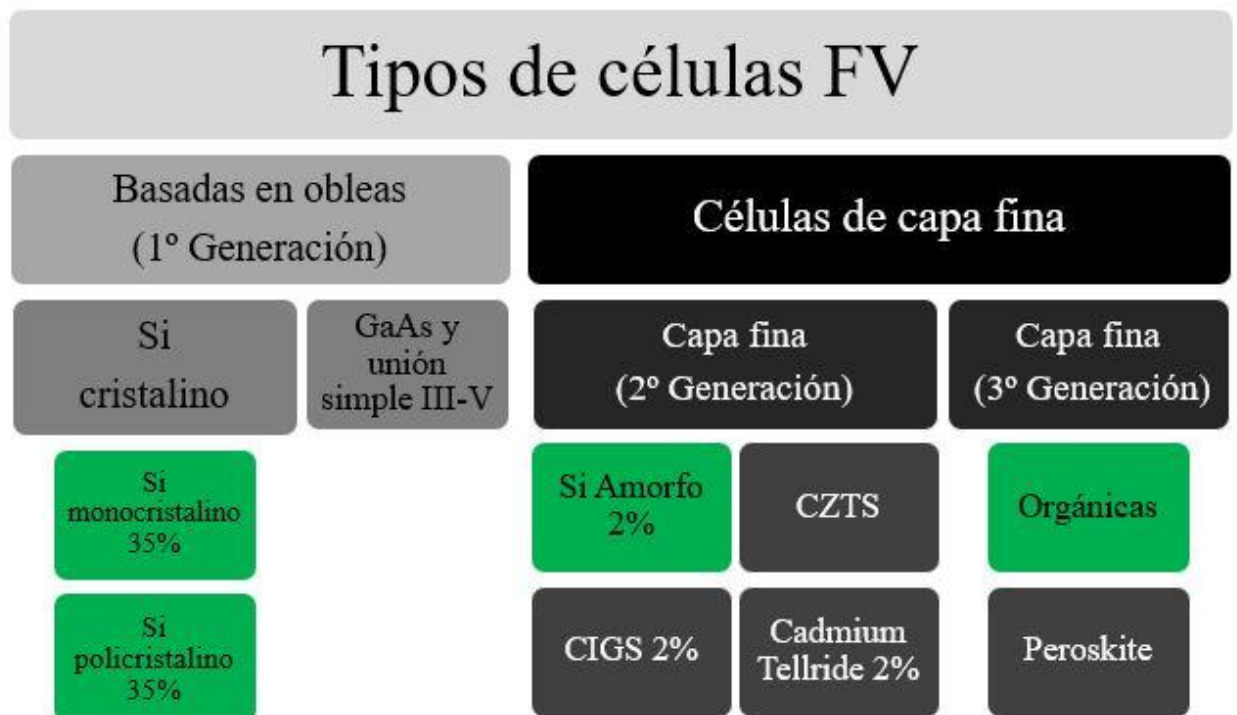
En cuanto a la **tercera generación** de células fotovoltaicas existen diferentes tecnologías, de las que los máximos exponentes son las **células orgánicas y las perovskitas**. Las células orgánicas se emplean en la electrónica orgánica para generar corriente eléctrica a través del efecto fotoeléctrico. La electrónica orgánica se basa en el uso de materiales polímeros orgánicos conductores o pequeñas moléculas orgánicas, una de las ventajas de esta tecnología es que permite una producción a gran escala a costes muy reducidos, pero presentan **baja eficiencia** y una **escasa resistencia** frente a las tecnologías de silicio cristalino.



**Figura 8.** Célula fotovoltaica orgánica.



En **el mercado actual** disponemos de una gran cantidad de diferentes materiales para crear células fotovoltaicas como podemos ver en la Figura 9 y las sombreadas en verde son las tecnologías que el alumno tendrá que caracterizar.



**Figura 9.** Esquema de las tecnologías en células fotovoltaicas.

Como podemos ver en el esquema de la Figura 9, la principal ramificación se produce entre las células de silicio y las de compuestos.

Dentro de la rama de las **células de Silicio** podemos diferenciar 3 tipos:

- **Células de Silicio metalúrgico (UMG):** alrededor del año 2008 algunas empresas empezaron a desarrollar esta tecnología debido a que el Silicio cristalino tenía un **precio** muy elevado por esa época. Actualmente esta tecnología **está en desuso** debido a que el precio del Silicio ha disminuido drásticamente y el rendimiento de las células fotovoltaicas de Silicio cristalino es mucho mayor.
- **Células de Silicio cristalino:** la mayor parte de las células fotovoltaicas instaladas actualmente son de este tipo. Como podemos observar en la Figura 9 se divide en 3 tipos a su vez:
  - **Células de silicio monocristalino**  
Son células que presentan una **estructura ordenada**, su comportamiento uniforme las hace buenas conductoras. Son difíciles de fabricar y las podemos reconocer por su monocromía azulada como podemos observar en la Figura 6.

- **Células de silicio policristalino**  
Son células que presentan una **estructura ordenada por regiones separadas**. Los enlaces irregulares disminuyen el rendimiento. Su superficie está estructurada en cristales con distintos tonos de azules y grises metálicos como podemos ver en la Figura 6.
- **Células de capa fina**  
Son células que presentan un **alto grado de desorden**, por su número de defectos estructurales y de enlaces. Por ello su **fabricación** es mucho más **simple** que otras tecnologías y sobre todo es mucho más **barato**. Son eficientes bajo iluminación artificial, radiación difusa o situaciones sombreadas. Podemos reconocerlas ya que son de color marrón homogéneo como podemos ver en la Figura 7.
- **Células de Silicio amorfo**: esta tecnología se viene aplicando desde hace años, se puede encontrar en cualquier **calculadora** alimentada por una célula solar. Su penetración en el mercado se reduce a un aproximado 2%. La fabricación de estas células es completamente distinta a las de Silicio cristalino, ya que en este caso se trabaja con **Silicio en estado gaseoso** que se deposita sobre un determinado sustrato, siendo este proceso mucho más sencillo que en el caso de la fabricación de las células de Silicio cristalino, pudiendo reducir los costes de producción de estas. Aunque las células de Silicio amorfo presentan una menor eficiencia, típicamente entre un 6-8% y además su vida útil es menor.

## 2.6 Células fotovoltaica: Instalaciones

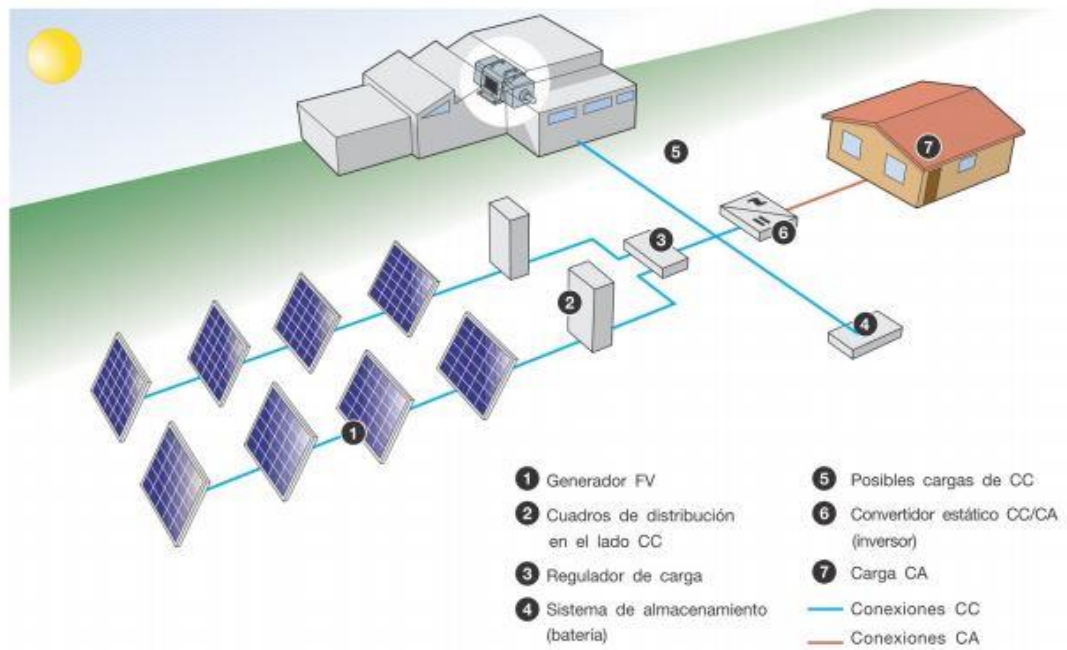
Todo lo descrito anteriormente corresponde con la parte fundamental de una instalación de energía solar, las células fotovoltaicas. Cuando juntamos muchos de estas células formamos un **módulo** y estos a su vez crean estructuras más grandes denominadas **instalaciones**, este tipo de instalaciones pueden ser de dos tipos :

- **Instalaciones aisladas**. Se trata de instalaciones diseñadas para alimentar, de manera **independiente** de la red eléctrica, instalaciones, infraestructuras o dispositivos. Como por ejemplo una vivienda de difícil acceso, una estación base de telefonía o una red de sensores, como podemos observar en la Figura 10.
- **Instalaciones conectadas a la red**. En este caso la energía generada se **inyecta** a la red eléctrica, actuando como una **central de generación**, como podemos observar en la Figura 11.

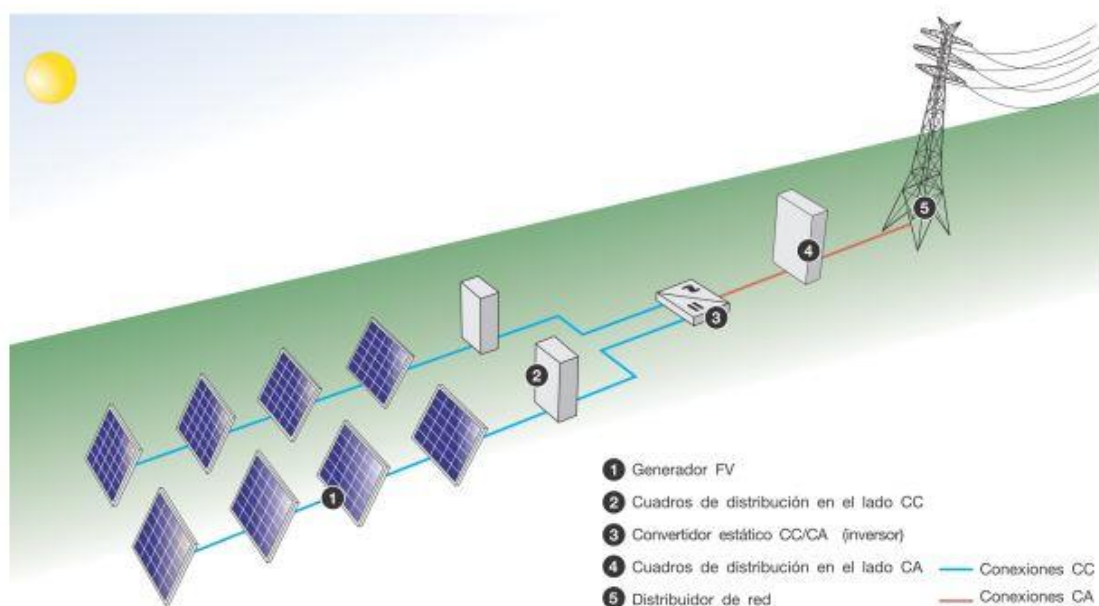
Los **elementos básicos** que componen las instalaciones conectadas a la red son los generadores fotovoltaicos, un inversor, una sección de distribución/protección eléctrica y un contador. Por otro lado, en una instalación aislada hay que considerar además un controlador/regulador de carga, así como baterías que forman un sistema de acumulación de energía.

- **Generador FV**. Se trata del conjunto de células fotovoltaicas montadas dentro de un panel fotovoltaico, y se encargar de generar la energía eléctrica que posteriormente será inyectada a la red o acumulado en un sistema de baterías.

- **Inversor.** Los paneles generan energía eléctrica en corriente continua, pero para su inyección a la red o para ciertos usos es necesario trabajar en corriente alterna. El inversor es el encargado de realizar la transformación DC-AC.
- **Regulador de carga.** En las instalaciones aisladas el regulador de carga se encarga de gestionar el flujo de energía. Por ejemplo, de los paneles a las baterías si hay radiación solar suficiente o de las baterías a las cargas si las baterías están cargadas.
- **Baterías.** En instalaciones aisladas las baterías permiten acumular energía para cuando los paneles no estén generando, en la noche o días nublados.



**Figura 10.** Esquema de una instalación aislada.



**Figura 11.** Esquema de una instalación conectada a la red.



## 3. Hardware open source

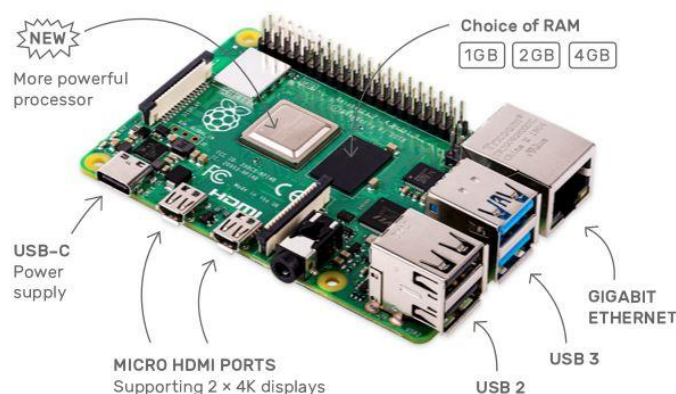
### 3.1 Introducción al hardware open source

El origen de los proyectos de hardware de “código abierto” se remonta a la época de los 70’s, aunque no es tan conocido como el software de “código abierto”. Los directores ejecutivos de las grandes empresas del Hardware y Software compartían de forma abierta todos sus diseños, proyectos, planos y PCBs. En los años 90 Reinoud Lamberts fue el primero en proponer la creación de una comunidad al estilo de la comunidad del Software libre. El software libre ofrece al usuario cuatro libertades: libertad de uso, de estudio y modificación, de distribución y de redistribución de las mejoras. El hardware open source toma estas mismas ideas para aplicarlas en su campo, pero **la distribución de hardware es mucho más complicada** que la distribución de software y sobre todo al principio por que no existía software para el diseño electrónico.

Unas de las primeras iniciativas fue el **proyecto Arduino**, que además será la base de nuestra práctica. Esta plataforma consiste en una placa con diversas **entradas y salidas** que permite el desarrollo de infinidad de proyectos y sirve como iniciación para estudiantes en el mundo de la electrónica y que en el siguiente apartado profundizaremos más en esta plataforma.

En torno a este concepto surgen otras plataformas de hardware open source como puede ser **Raspberry Pi**, que se trata de un mini pc del tamaño de una tarjeta de crédito, en el que nos encontramos con un **procesador, un chip gráfico y memoria RAM**. En el año 2009 se construye la Fundación Raspberry Pi en Caldecote, South Cambridgeshire, Reino Unido como una **asociación caritativa** que es regulada por la comisión de caridad de Inglaterra y Gales. Eben Upton, se puso en contacto con un grupo de profesores, académicos y entusiastas de la informática para crear un ordenador con la intención de animar a los niños a aprender informática y crean el primer prototipo basado en ARM ocupando el mismo tamaño que una memoria USB.

En la actualidad podemos encontrar 2 tipos de modelos los “A” y los “B”, la diferencia entre ambos es simplemente que el modelo A tiene unas especificaciones peores a un menor precio, por ejemplo, la diferencias entre los modelos Raspberry Pi 3 modelo B+ y Raspberry Pi 3 modelo A+ residen en un procesador menos potente, menos memoria RAM y menos puertos USB 2.0.



**Figura 12.** Raspberry Pi 4 Fuente: <https://www.raspberrypi.org/>

También podemos destacar **RepRap** que fue un proyecto iniciado en 2004 por Adrian Bowyer y sus compañeros de la Universidad de Bath, en Reino Unido, para crear una **impresora 3D capaz de replicarse a sí misma** mediante la impresión de la mayoría de sus componentes. El dispositivo está accesible bajo licencia GNU GPL, que permite que cualquiera pueda copiar, estudiar, distribuir y mejorar sus diseños y código fuente. Gracias a ello, existen varios modelos de RepRap que día a día son mejorados por la comunidad de hardware libre.

Y por último podemos destacar el proyecto **Uzebox** que tenía como objetivo desarrollar una **consola de videojuegos** totalmente libre y abierta, un dispositivo que se ha distribuido en forma de kit y que permite rescatar los mandos de una "clásica consola" como podemos ver en la Figura 13.

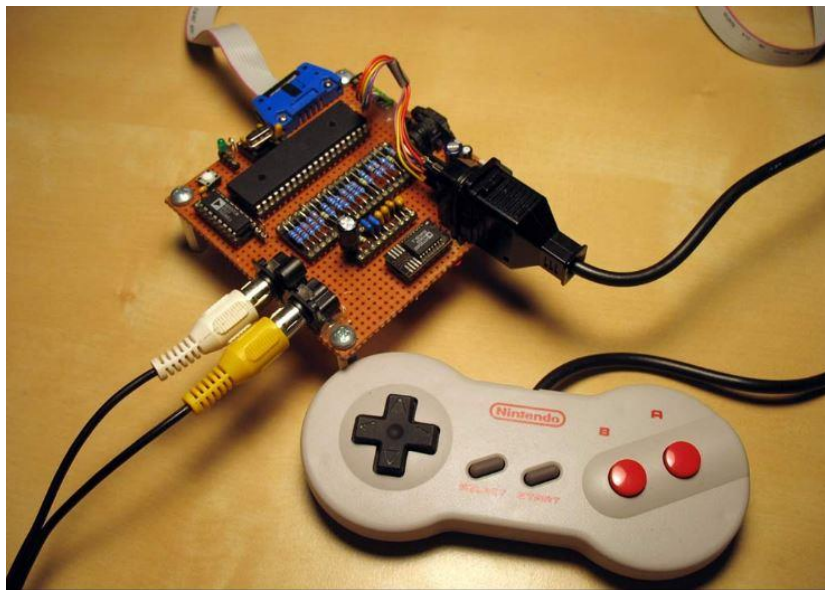


Figura 13. Uzebox Fuente: <http://belogic.com/uzebox/index.asp>

## 3.2 Introducción a Arduino

El proyecto Arduino fue desarrollado en el año 2005 por el entonces estudiante del instituto de Diseño Interactivo (IVRAE) Massimo Banzi en Italia, que en un principio pensaba en hacer Arduino por una necesidad de **aprendizaje para los estudiantes** de computación y electrónica del mismo instituto, ya que por aquella época adquirir una placa con un microcontrolador era bastante caro y además no ofrecían un soporte adecuado. Inicialmente fue un proyecto creado no solo para economizar la creación de proyectos escolares dentro del instituto, sino que, además, Banzi tenía la intención de ayudar a su escuela a evitar la quiebra de esta con las ganancias que produciría vendiendo sus placas dentro del campus a un precio accesible (1 euro por unidad).

Con el paso del tiempo Arduino ha generado una legión de seguidores debido a su sencillez y la multitud de tareas que puede ejecutar esta placa. Resumiendo, estas placas utilizan un microcontrolador **Atmel AVR** para dirigir los puertos entrada/salida y un microcontrolador de la familia **ATMEGA** para dirigir el flujo de datos.

Hoy en día Arduino **es la herramienta más vendida mundialmente** para proyectos DIY (*Do It Yourself*). Este éxito de Arduino se debe a los siguientes factores:

- **Asequible:** las placas Arduino son relativamente baratas en comparación con otras plataformas de microcontroladores. La versión menos costosa del módulo Arduino puede ser montada a mano.
- **Multiplataforma:** el software de Arduino funciona en sistemas operativos Windows, Macintosh OSX y Linux.
- **Entorno de programación simple y claro:** el entorno de programación de Arduino es suficiente para los usuarios avanzados y de fácil uso para los principiantes.
- **Software:** se publica como herramientas de código abierto.
- **Hardware:** el Arduino se basa en ATMEGA8 y ATMEGA168 microcontroladores de Atmel. Los planos para los módulos están publicados bajo una licencia de Creative Commons, por lo que los diseñadores de circuitos con experiencia pueden hacer su propia versión del módulo, ampliándolo y mejorándolo.

### 3.3 Tipos de Arduino en el Mercado

En la actualidad existen varios modelos de Arduino, algunos de los principales son:

#### - Arduino UNO Rev3

Características:

- |                        |                       |
|------------------------|-----------------------|
| ○ Microcontrolador     | → ATmega328P          |
| ○ Operación de Voltaje | → 5 V                 |
| ○ Entradas Analógicas  | → 6                   |
| ○ Entradas Digitales   | → 14 (6 entradas PWM) |
| ○ Memoria Flash        | → 32 KB               |



Figura 14. Placa Arduino UNO Fuente: <https://www.arduino.cc/>

### - Arduino DUE

#### Características:

- Microcontrolador → AT91SAM3X8E
- Operación de Voltaje → 3.3 V
- Entradas Analógicas → 12
- Entradas Digitales → 54 (12 entradas PWM)
- Memoria Flash → 512 KB

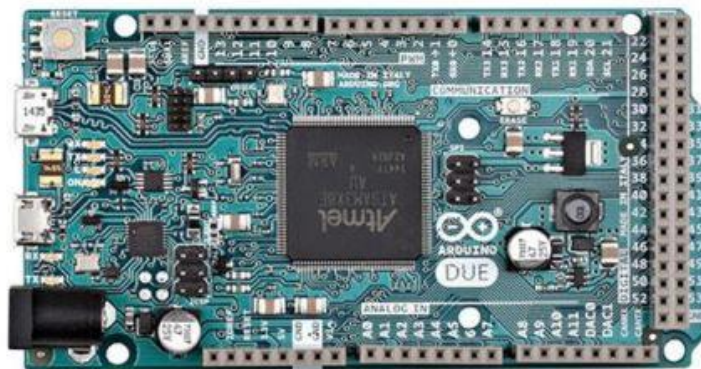


Figura 15. Placa Arduino DUE Fuente: <https://www.arduino.cc>

### - Arduino ZERO

#### Características:

- Microcontrolador → ATSAMD21G18, 32-Bit ARM
- Operación de Voltaje → 3.3 V
- Entradas Analógicas → 6
- Entradas Digitales → 20
- Memoria Flash → 256 KB



Figura 16. Placa Arduino ZERO Fuente: <https://www.arduino.cc/>



### - Arduino MEGA 2560 Rev3

#### Características:

- Microcontrolador → ATmega2560
- Operación de Voltaje → 5 V
- Entradas Analógicas → 16
- Entradas Digitales → 54 (15 entradas PWM)
- Memoria Flash → 256 KB

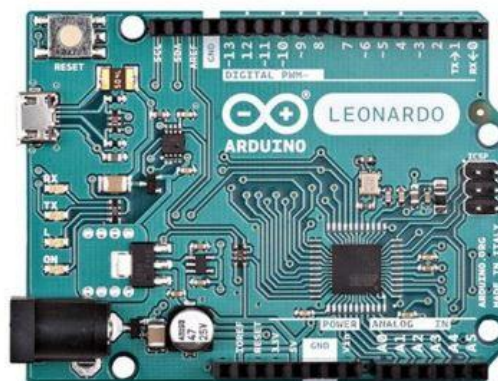


**Figura 17.** Placa Arduino MEGA 2560 Rev3 Fuente: <https://www.arduino.cc>

### - Arduino LEONARDO

#### Características:

- Microcontrolador → ATmega32u4
- Operación de Voltaje → 5 V
- Entradas Analógicas → 12
- Entradas Digitales → 20 (7 entradas PWM)
- Memoria Flash → 32 KB



**Figura 18.** Placa Arduino LEONARDO Fuente: <https://www.arduino.cc/>

En el desarrollo de la práctica no hemos decantado por **usar la placa Arduino LEONARDO**, debido a que es una de las placas **más actuales** y tiene suficientes entradas para nuestra práctica. En nuestro caso podríamos usar cualquiera de las placas anteriormente nombradas ya que el número máximo de sensores usados es de 3 como veremos en el **capítulo 4 Componentes del sistema**.

### 3.4 *Entorno de desarrollo de Arduino*

El entorno de desarrollo en Arduino (IDE) es el encargado de controlar la comunicación entre el PC y el hardware de Arduino. Se establece la conexión entre ambos a través de los puertos USB de estos dispositivos. Como podemos ver en la Figura 19, el entorno de desarrollo de Arduino se compone de:

- **Editor de texto**

Este espacio está destinado para escribir el **código** para Arduino, este entorno de desarrollo permite copiar, pegar, buscar y remplazar texto. El código de Arduino como podemos ver en la Figura 19 está dividido en 2 partes bien diferenciadas.

El **void setup()** es la primera función en ejecutarse dentro de un programa en Arduino, aquí se describen los parámetros fundamentales como por ejemplo si nuestro programa va a usar comunicación serial. También en este apartado tenemos que definir los pines que vamos a usar dentro de Arduino y su comportamiento, pines de entrada o de salida.

El **void loop()** es la función que se ejecuta un número infinito de veces hasta que se apague o se reinicie el microcontrolador.

Arduino primero ejecuta una vez el setup y después entra en bucle en esta parte.

- **Área de mensajes**

Área donde el usuario puede comprobar toda la información mientras se cargan los programas, como los posibles **errores o advertencias** a la hora de compilar debido a problemas en el código o por fallo en la detección de nuestro Arduino en el puerto USB. Una vez terminada la compilación también muestra información sobre el **porcentaje de memoria ocupada** por nuestro programa en el microcontrolador.

- **Barra de herramientas**

Barra donde se encuentran los **accesos directos** fundamentales dentro de Arduino. Dentro de esta barra de herramientas podemos encontrar las siguientes funciones:



Este acceso directo permite **verificar el código del editor de texto**, realiza una compilación del código y en caso de detectar problemas en el código, a través del área de mensajes notificará con dichos errores.



Este acceso directo realiza el mismo proceso que el descrito en el anterior punto, pero si después de verificar el código no hay errores, **carga el programa en Arduino**.



Este acceso directo nos permite **crear un nuevo documento** en el editor de texto.



Este acceso directo nos permite **abrir un archivo** en el editor de texto.



Este acceso directo nos permite **guardar el fichero** en la ruta que seleccionemos, este fichero será guardado en una carpeta con el mismo nombre de forma automática.



Este acceso directo permite **acceder al puerto serie** entre el PC y Arduino. Al pinchar sobre este acceso directo el entorno de desarrollo de Arduino abre una nueva ventana a través de la cual se puede ver la comunicación entre el PC y Arduino, en esta comunicación podemos enviar caracteres a Arduino y viceversa.

La principal funcionalidad del monito serial es que Arduino **nos envía datos** a través de este puerto. En nuestra práctica Arduino nos devuelve datos de sensores a través del puerto serie.



**Figura 19.** Entorno de desarrollo Arduino.

## 4. Caracterización de células FV

### 4.1 Característica I-V

Una célula fotovoltaica es una **unión PN** que se comporta como un **diodo** cuando no recibe ningún tipo de radiación y, por tanto, la corriente está determinada por el modelo matemático de **Shockley**. Sin embargo, al recibir radiación, a la corriente determinada por el modelo de Shockley, se le tiene que sumar otra corriente, la **corriente fotogenerada** ( $I_f$ ) como podemos ver en la siguiente ecuación :

$$I = I_s \left( e^{\frac{V_d}{n \cdot V_t}} - 1 \right) - I_f = I_d - I_f$$

$$V_t = \frac{k \cdot T}{q}$$

Donde:

- $I_d$  = Corriente diodo.
- $I_s$  = Corriente de saturación inversa.
- $I_f$  = Corriente fotogenerada.
- $V_d$  = Tensión aplicada al diodo.
- $V_t$  = Tensión de temperatura.
- $k$  = constante de Boltzmann.
- $q$  = Carga elemental del electrón.
- $T$  = Temperatura en Kelvin.
- $n$  = constante de difusión (Ge=1, Si=2).

#### Ecuación 1. Modelo matemático Shockley

La corriente fotogenerada ( $I_f$ ) se debe a que, al iluminar la célula fotovoltaica, los fotones que inciden con una **energía igual o superior** al ancho de la banda prohibida generan pares de electrones-huecos que son separados por el campo eléctrico.

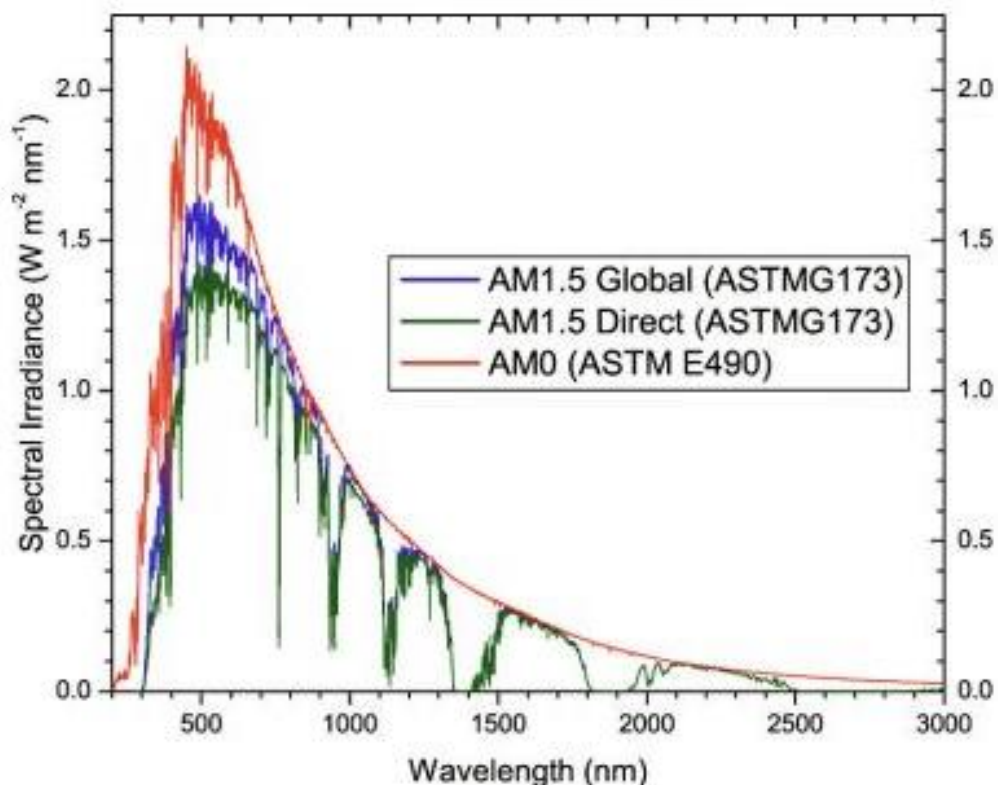
Esta corriente fotogenerada ( $I_f$ ) tiene el efecto de **desplazar la curva I-V** al cuarto cuadrante de cada célula fotovoltaica, donde se genera energía y, por tanto, equivale a un modelo ideal de una **fuentes de corriente**. El comportamiento de la célula fotovoltaica en la **oscuridad** coincide con la de un **diodo**, ya que la principal diferencia entre ambos es el contacto metálico superior que en el caso de la célula fotovoltaica presenta forma de rejilla, diseñada de tal forma que la luz pueda atravesar en el semiconductor.

Cuando caracterizamos una célula fotovoltaica dependemos de **circunstancias externas** a la propia célula, como son la temperatura, irradiancia o el espectro del sol.



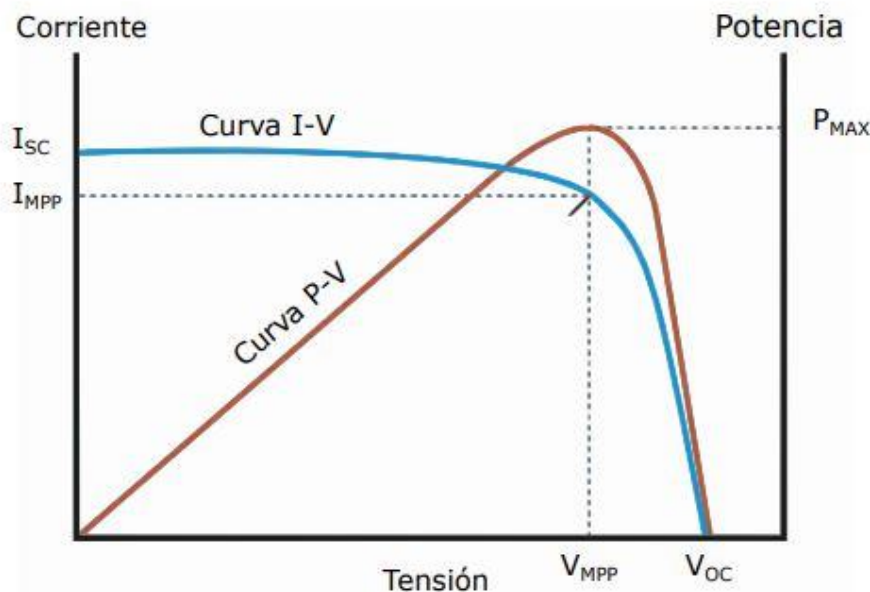
Para que los fabricantes caractericen sus células fotovoltaicas de **una manera eficiente** y fiel deben seguir las **Condiciones Estándar de Medida** (CEM). Estas condiciones definen los valores de los parámetros clave con los que se deben caracterizar los parámetros eléctricos de cualquier célula fotovoltaica, como son:

- **Irradiancia:** Se define como la **potencia por unidad de superficie** asociada a una radiación electromagnética dada. La irradiancia que se define en las CEM es de  $1000 \text{ W/m}^2$ .
- **Temperatura de la célula:** La temperatura de la célula afecta considerablemente al rendimiento de la célula fotovoltaica como veremos en el capítulo 5.2, por ello es muy importante definir claramente una temperatura de trabajo para caracterizar una célula fotovoltaica, según **la CEM es de  $25^\circ\text{C}$** .
- **Espectro solar:** Las células fotovoltaicas no se caracterizan con la luz natural del sol, sino con iluminación artificial por eso es necesario definir el espectro de la fuente de luz con el que se realizarán las medidas. Este espectro es el **AM1.5** situado en la Figura 20.



**Figura 20.** Espectro solar. Fuente: <https://www.pveducation.org>

Todos los valores eléctricos a la hora de caracterizar una célula fotovoltaica tienen que seguir las CEM, los cuales definirán **las curvas I-V y P-V** como podemos ver en la Figura 21.



**Figura 21.** Característica I-V y P-V de una célula FV. Fuente: <https://bit.ly/2wZ3Odu>

**V<sub>MPP</sub>, Tensión para máxima potencia.** Es la tensión que entrega la célula a potencia máxima bajo las CEM. Se suele usar como tensión nominal de la célula fotovoltaica.

**I<sub>MPP</sub>, Corriente para máxima potencia.** Es la corriente que entrega la célula a potencia máxima bajo las CEM.

**P<sub>MAX</sub>, Punto de Máxima Potencia.** Es la máxima potencia eléctrica que puede entregar un panel bajo las CEM. En este punto es donde la célula fotovoltaica entrega la **tensión y corriente máxima**.

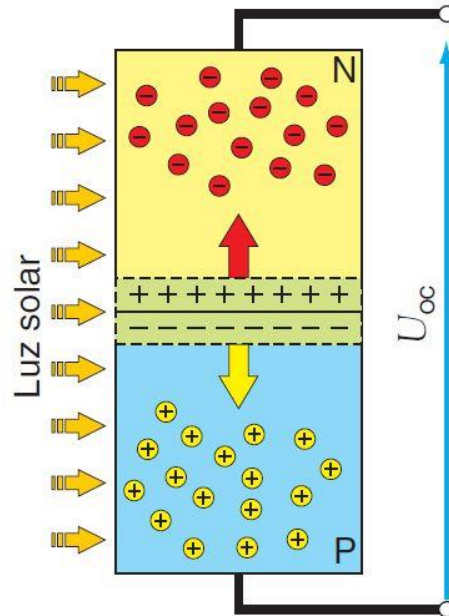
$$P_{MAX} = V_{MPP} * I_{MPP}$$

**FF, Factor de forma.** El factor FF es un número sin dimensiones que cuantifica la relación que hay entre la potencia máxima y la potencia que generaría una célula fotovoltaica hipotética cuya curva I-V fuera un rectángulo perfecto de lados I<sub>sc</sub> y V<sub>oc</sub>.

**η, Eficiencia de conversión.** La eficiencia o rendimiento de conversión es el porcentaje de potencia convertida en energía eléctrica capaz de hacer funcionar un circuito. Se define como el cociente entre la potencia máxima entregada por la célula fotovoltaica y la potencia recibida del sol.

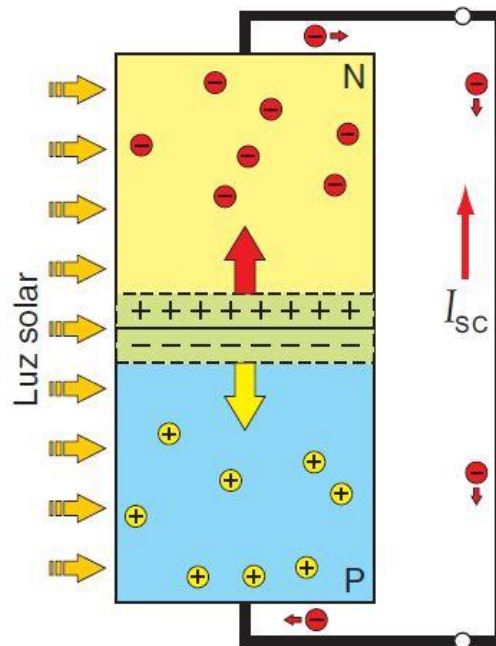
**SR, Respuesta espectral.** Es el cociente entre la densidad de corriente de cortocircuito y la densidad espectral de potencia incidente para una longitud de onda determinada. Esta relación nos indica que longitudes de onda son más eficientes para nuestra célula fotovoltaica.

**Voc, Tensión en circuito abierto.** Es la diferencia de potencial que se alcanza cuando una célula fotovoltaica es iluminada, sin estar en contacto las regiones P y N, siendo proporcional a la iluminación recibida. Es la máxima tensión que puede entregar una célula fotovoltaica a una corriente nula bajo las CEM.



**Figura 22.** Célula fotovoltaica en circuito abierto.

**Isc, Corriente en cortocircuito.** Es la corriente que se genera cuando las regiones P y N están unidas por un conductor exterior con una resistencia nula y es proporcional a la iluminación recibida. Es la máxima corriente que puede entregar una célula a corriente nula, en las CEM.

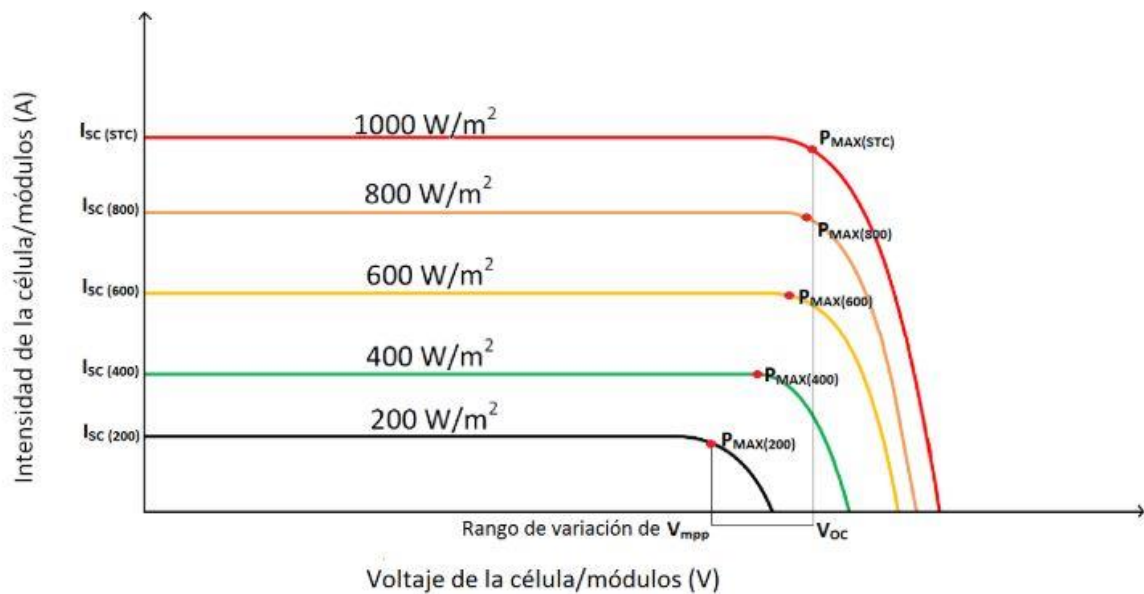


**Figura 23.** Célula fotovoltaica en cortocircuito.

## 4.2 Influencia de la irradiancia

Como hemos visto en el capítulo anterior, las curvas I-V que los fabricantes nos proporcionan de las células fotovoltaicas son realizadas con las CEM.

Estas curvas concretamente están obtenidas para  $1000 \text{ W/m}^2$ , pero la tensión y la corriente generada en una célula fotovoltaica depende directamente de la irradiancia recibida. La corriente de cortocircuito de la célula es **directamente proporcional** a la irradiancia como se puede ver en la Figura 24, disminuyendo a media que se reduce la irradiancia. La tensión de circuito abierto varía poco con respecto a la irradiancia, a efectos prácticos se puede considerar constante.



**Figura 24.** Influencia de la irradiancia en células fotovoltaicas.

La intensidad de corto circuito como podemos ver varía según la irradiación, siendo una **variación lineal** siguiendo la siguiente ecuación:

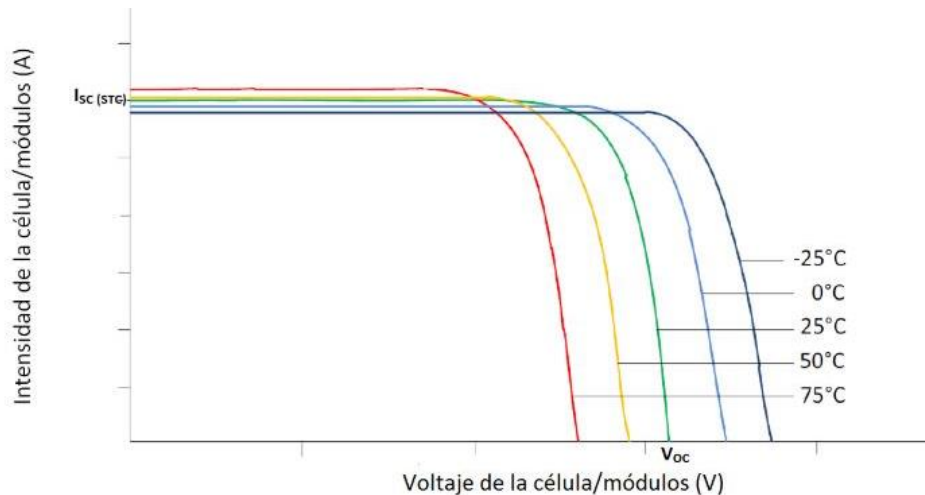
$$I_{sc}(G) = G * \left( \frac{I_{sc}(CEM)}{1000} \right)$$

Donde:  $I_{sc}(G)$  = Corriente de cortocircuito para una irradiancia  $G$   
 $I_{sc}(CEM)$  = Corriente de cortocircuito en las CEM\*  
 $G$  = Irradiancia ( $\text{W/m}^2$ )

\*Condiciones Estándar de Medida

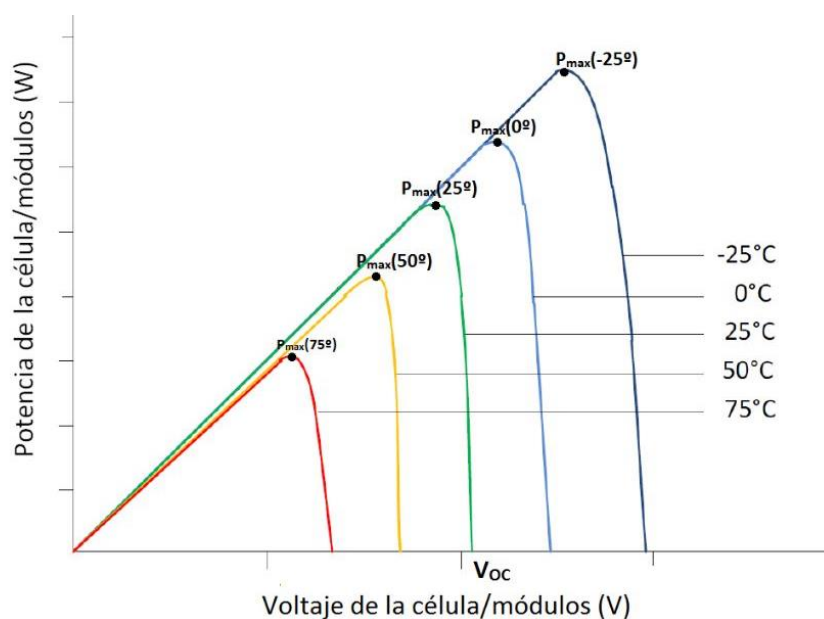
### 4.3 Influencia de la temperatura

Las células fotovoltaicas **funcionan mejor en climas fríos**, siempre y cuando no se superen cierto límites, esto se debe a que las células fotovoltaicas son dispositivos electrónicos y como todos los dispositivos electrónicos su **rendimiento se ve afectado** por el aumento de las temperaturas de estos. Por esta razón la temperatura afecta de manera considerable a la tensión como podemos observar en la Figura 25.



**Figura 25.** Influencia de la temperatura en células fotovoltaicas.

Como se aprecia la **tensión de circuito abierto disminuye** cuando aumenta la temperatura. La **intensidad de cortocircuito** sin embargo, **aumenta** cuando la temperatura aumenta, aunque la variación es muy pequeña y a efectos prácticos puede considerar constante. Es evidente que si la tensión de la célula disminuye cuando aumenta la temperatura y la intensidad prácticamente se mantiene constante, la potencia entregada por la célula disminuirá cuando aumente la temperatura de la célula como podemos ver en la Figura 26.



**Figura 26.** Variación de la potencia entregada respecto a la temperatura.

La **temperatura de trabajo** de una célula fotovoltaica está relacionada con la **temperatura ambiente** y la **irradiancia** como podemos observar en la siguiente ecuación:

$$T_c = T_A + G * \left( \frac{T_{ONC} - 20}{800} \right)$$

Donde:  $T_c$  = Temperatura de trabajo de la célula °C  
 $T_A$  = Temperatura ambiente °C  
 $T_{ONC}$  = Temperatura de operación nominal de la célula °C\*  
 $G$  = Irradiancia ( $W/m^2$ )

\*Corresponde a una irradiación en el plano de la célula de  $800 W/m^2$  con orientación a la radiación incidente al mediodía solar, **temperatura ambiente de 20°C**, **velocidad del viento de 1 m/s** y **funcionamiento en circuito abierto**.

Para concluir podemos afirmar que, a mayor temperatura, la célula fotovoltaica presenta un menor voltaje, menor potencia generada y mayor intensidad de cortocircuito. Esta variación respecto a la temperatura se expresa también mediante los coeficientes:

- **Coefficiente intensidad-temperatura** ( $\alpha$ ): Se expresa en valor absoluto en mA/°C. Un valor típico en células de silicio es de 0,63mA/°C.
- **Coefficiente tensión-temperatura** ( $\beta$ ): Se expresa en valor absoluto en %/°C. Un valor típico en células de silicio es de -0,33%/°C.



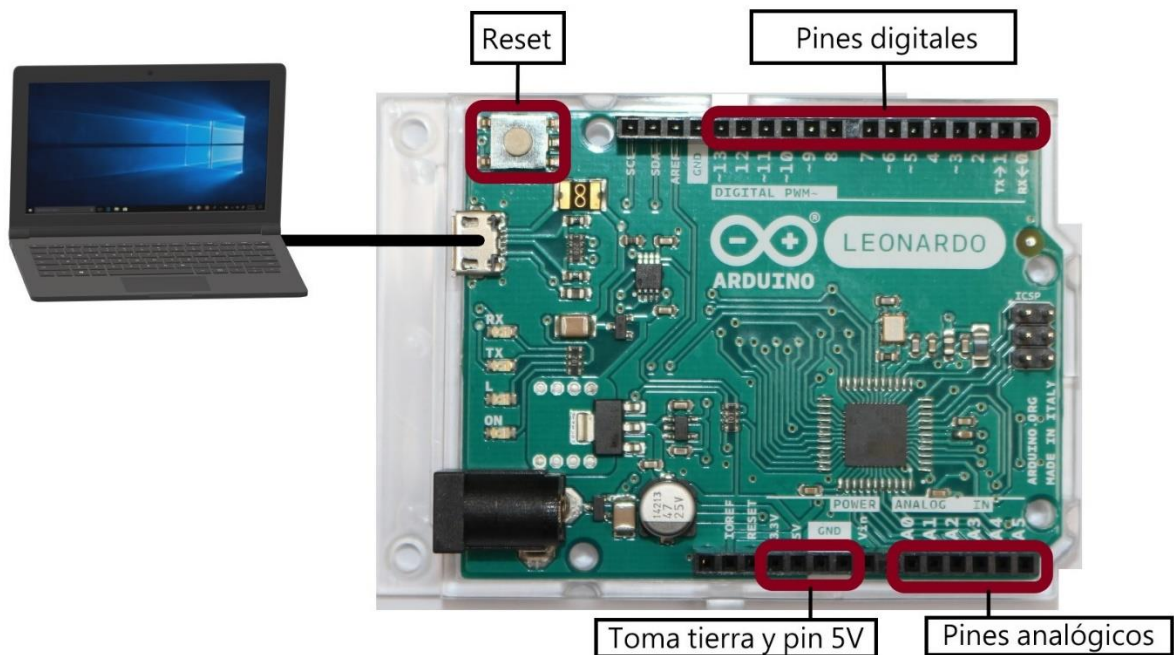
## 5. Componentes del sistema

### 5.1 Placa Arduino Leonardo

La placa Arduino es una pieza fundamental de este proyecto, debido a que esta placa es una primera toma de contacto entre el alumno y el mundo del **hardware open source**. Es una placa basada en el **microcontrolador ATmega32u4**, este microcontrolador tiene incorporada la **comunicación USB**, lo que elimina la necesidad de un coprocesador encargado del protocolo de comunicación. Tiene **12 pines de entradas/salidas digitales** de los cuales 7 se pueden usar como salidas PWM (modulación por ancho de pulsos) como podemos ver en la Figura 27, nos tenemos que fijar que estas entradas/salidas digitales PWN tienen el símbolo  $\sim$ . En cuanto a las **entradas analógicas** podemos observar en la Figura 27 que nuestra placa Arduino Leonardo posee **6 pines analógicos**.

En versiones anteriores ya que era necesarios el uso de un coprocesador para el protocolo de comunicación se necesitaba alimentar la placa Arduino de una forma externa al puerto USB. En nuestro caso se alimenta la placa a través **del puerto USB** de nuestro ordenador. El interfaz USB es micro-USB, características de la mayoría de smartphone comercializados durante los últimos años. Necesitaremos un cable con un extremo USB-A (ordenador) y el otro extremo micro-USB (Arduino).

Por último, cabe destacar que Arduino implementa un botón para realizar un **reset** de la placa y eliminar la implementación que tenga en ese momento Arduino, para aquellas ocasiones en las que la placa se quede bloqueada.



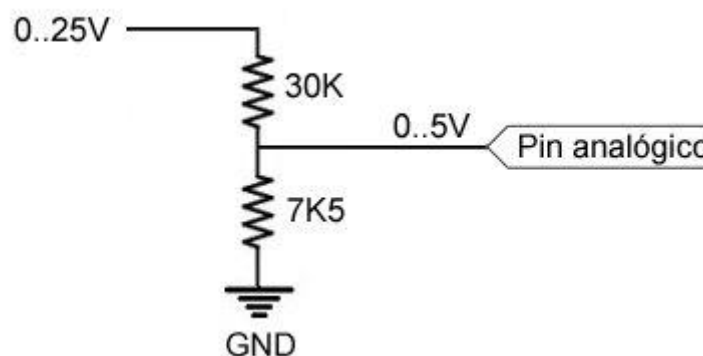
**Figura 27.** Placa Arduino Leonardo.

En la práctica queremos realizar una **caracterización de la respuesta eléctrica** de un conjunto de células fotovoltaicas, para ello es necesario el uso de varios sensores para obtener datos de la **corriente, voltaje y temperatura**. A continuación, se detallarán las **características, esquema y códigos necesarios** para que alumno se adentre en el mundo de la energías renovables a la vez que aprende a usar todos los recursos disponibles del hardware open-source, siendo este último un objetivo muy importante en la práctica.

## 5.2 Sensor voltaje FZ0430

El sensor FZ0430 es un módulo comercial que nos permite **medir tensiones de hasta 25V** de una forma sencilla con nuestra placa Arduino.

Se trata de un **divisor de tensión** con resistencias de valor 30k $\Omega$  y 7,5k  $\Omega$ .



**Figura 28.** Divisor de tensión sensor FZ0430

El sensor de voltaje FZ0430 dispone de una clema de conexión, para conectar la carga que deseemos medir. En la otra parte del sensor disponemos de **tres pines** para la interconexión con la placa Arduino, como podemos ver en la Figura 29.

- S: pin digital de lectura de la señal
- +: alimentación a 5V
- -: conexión a tierra

Las principales **características técnicas** de este sensor son las siguientes:

- Alimentación  $\rightarrow$  5 V
- Voltaje detección máxima  $\rightarrow$  25 V
- Voltaje detección entrada mínimo  $\rightarrow$  24,45 mV
- Resolución analógica de tensión  $\rightarrow$  0,00489 V DC

La conexión de este sensor con Arduino es trivial como podemos ver en la Figura 29. Una vez en IDE de Arduino tenemos que realizar la **lectura del pin analógico**. Para ello, declaramos una constante de valor entero para asignar el pin en la placa Arduino y otra constante de tipo decimal para devolver el valor de lectura del pin analógico, como podemos ver en la parte superior del código de la Figura 30. Para la lectura del pin analógico tan solo tenemos que llamar al método `analogRead()`, el cual está incluido en la librería por defecto de Arduino.



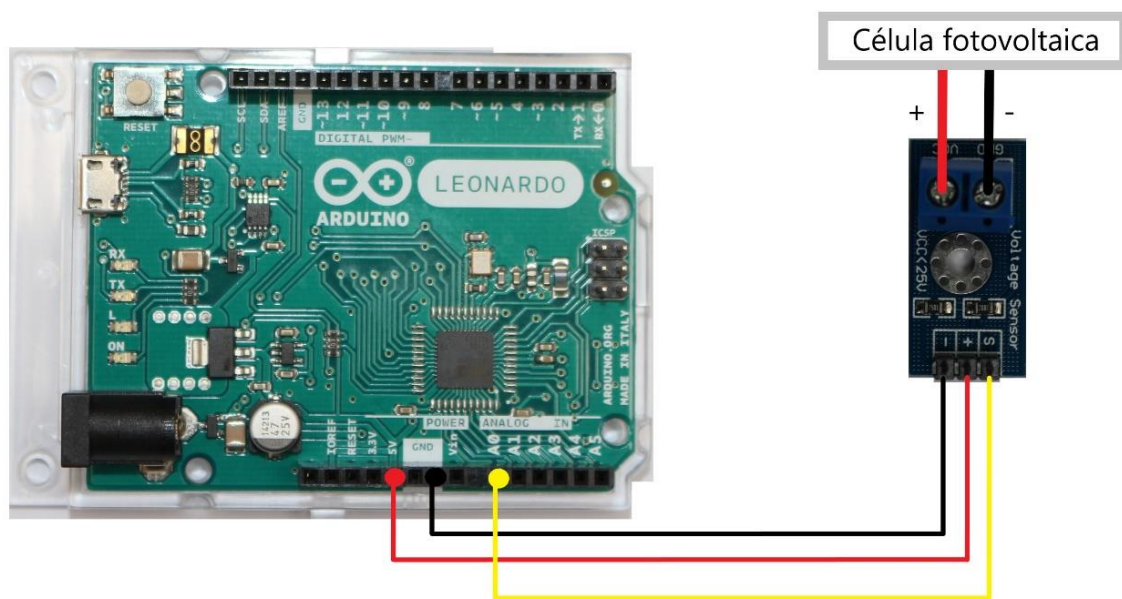


Figura 29. Conexión entre sensor FZ0430 y Arduino

```

VOLTAJE Arduino 1.8.12
Archivo Editar Programa Herramientas Ayuda

VOLTAJE

const int PinA0Voltaje = A0;
int sensorVoltajeLectura;
float voltaje;

void setup() {
  Serial.begin(9600);
}

void loop() {
  sensorVoltajeLectura = analogRead(PinA0Voltaje);
  voltaje = cambiarEscala(sensorVoltajeLectura, 0, 1023, 0.0, 25.0);
  Serial.println(voltaje);
  delay(1000);
}

float cambiarEscala(float x, float in_min, float in_max, float out_min, float out_max) {
  return (x - in_min)*(out_max - out_min)/(in_max-in_min)+out_min;
}
  
```

Figura 30. Código implementación IDE Arduino.

### 5.3 Sensor corriente ACS712

El sensor de corriente ACS712 internamente posee un **sensor de efecto Hall**, el cual **detecta el campo magnético** que se produce por inducción de la corriente que circula por la línea que se está midiendo en ese momento.

El sensor nos **devuelve** un valor de **2,5 V para una corriente de 0A** y a partir de este valor incrementa proporcionalmente de acuerdo con la sensibilidad del sensor, teniendo una relación lineal entre la salida de voltaje del sensor y la corriente.

Esta **relación lineal** se corresponde con la siguiente ecuación:

$$V = mI + 2,5$$

V (voltaje), m (sensibilidad sensor) e I(corriente)

Para facilitar la tarea de programación que abordaremos en un capítulo posterior, tenemos que despejar la intensidad de la ecuación anterior.

$$I = \frac{V - 2,5}{m}$$

V (voltaje), m (sensibilidad sensor) e I(corriente)

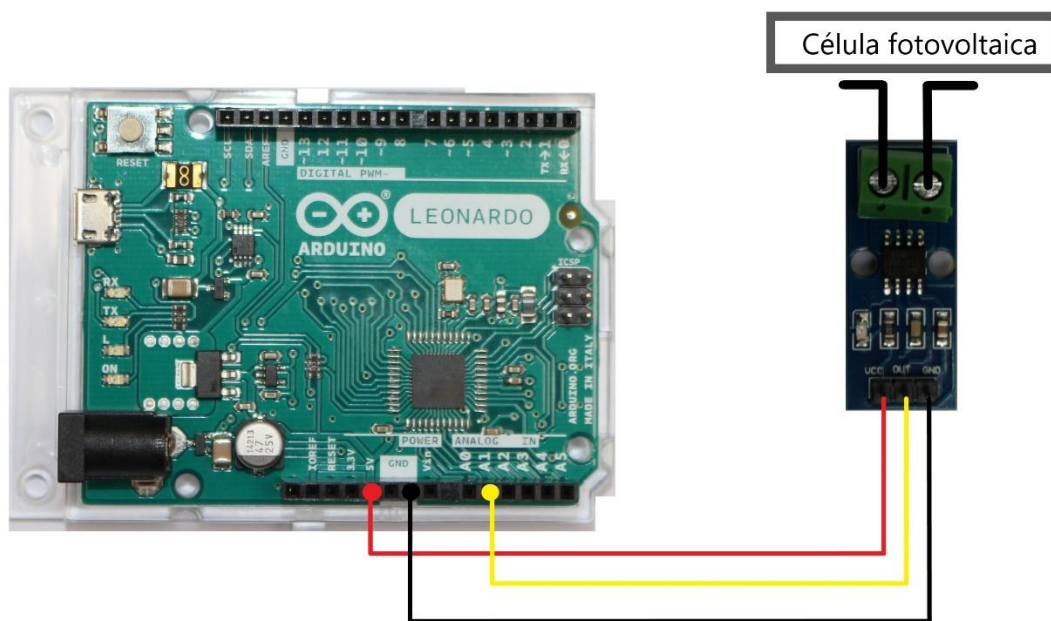
El sensor de Corriente dispone de una clema de conexión, para conectar la carga que deseamos medir. En la otra parte del sensor disponemos de **tres pines** para la interconexión con la placa Arduino, como podemos ver en la Figura 31.

- VCC: alimentación a 5V
- OUT: pin digital de lectura de la señal
- GND: conexión a tierra

Las principales **características técnicas** de este sensor son las siguientes:

- Alimentación → 5 V
- Sensibilidad → 185 mV/A
- Ruido → 21 mV
- Resistencia interna → 1,2 mΩ

La conexión de este sensor con Arduino es trivial como podemos ver en la Figura 31. Una vez en IDE de Arduino tenemos que declarar una **constante de tipo decimal** para introducir la sensibilidad de nuestro sensor. Primero tenemos que realizar la lectura del voltaje del sensor para después a través de la ecuación anterior calcular la corriente que atraviesa el sensor.



**Figura 31.** Conexión entre sensor ACS712 y Arduino

```

INTENSIDAD Arduino 1.8.12
Archivo Editar Programa Herramientas Ayuda

float Sensibilidad=0.185;

void setup() {
  Serial.begin(9600);
}

void loop() {
  float corrienteSensor= analogRead(A1)*(5.0 / 1023.0);
  float I=(corrienteSensor-2.5)/Sensibilidad;

  Serial.print(I*100);
  Serial.println(" mA");
  delay(1000);
}
  
```

16 Arduino Leonardo en COM5

**Figura 32.** Código implementación IDE Arduino.

El **error en las medidas** que obtenemos con la sensibilidad de 185mV/A puede ser un factor clave en la lectura de datos de este sensor de corriente. Los valores que nos facilita el fabricante no son exactos debido a que **cada sensor tiene unas características distintas** entre los sensores de una misma serie. Teniendo en cuenta esto, es un factor clave realizar una **calibración del sensor** para obtener una medida más precisa en nuestra práctica.

Recordemos la ecuación usada para obtener la corriente:

$$I = \frac{V - 2,5}{m}$$

En esta ecuación solo tenemos **dos constantes**, 2.5 que es el voltaje del sensor cuando la corriente es de 0A y la sensibilidad (m) que equivale a la pendiente de la recta.

Para calibrar el sensor tenemos que calcular estas dos constantes de nuevo.

Para ello, necesitamos medir la corriente con un amperímetro y realizar una captura de datos en dos puntos distintos, se recomienda obtener el punto en el que la corriente es 0A y el otro punto en 1A.

$$\text{Sensibilidad} = \frac{V2 - V1}{I2 - I1}$$

Donde:      V1 = Voltaje para 0A  
                  V2 = Voltaje para 1A  
                  I1 = Corriente de 0A  
                  I2 = Corriente medida por amperímetro para 1ª

Con la **nueva sensibilidad** calculada ya podemos trabajar con el menor error posible dentro de nuestra práctica, un factor importante para la caracterización de una célula fotovoltaica.

## 5.4 Sensor temperatura DHT11

El sensor de temperatura DHT11 es un dispositivo que se comunica de **forma digital**. Este sensor lo podemos encontrar en el mercado de manera individual o insertado en un PCB. La **versión con PCB** tiene incorporada una **resistencia pull-up de 5KΩ** y un LED que nos avisa de su funcionamiento.

En nuestra practica hemos implantado la versión con PCB, esta versión dispone de **tres pines** los cuales son:

- GND: conexión con tierra
- DATA: transmisión de datos a la placa Arduino
- VCC: alimentación a 3,3V

Las principales **características técnicas** de este sensor son las siguientes:

- Alimentación → 3,3V a 5 V
- Consumo → 2,5 mA
- Señal de salida → digital
- Rango temperatura → 0°C a 50 °C
- Precisión → a 25°C ± 2°C

El sensor de temperatura DHT11 **permite medir la temperatura y la humedad** del ambiente, en nuestro caso solo realizaremos medidas de la temperatura.

Es un sensor digital pero que la toma de datos del ambiente se realiza de forma analógica y dentro de nuestro sensor DHT11 se hace la **conversión** entre analógica y digital. Por lo tanto, partimos de una señal analógica que luego se convierte en formato digital y se envía al microcontrolador. La trama de datos es de **40 bits** que corresponde a la información de humedad y temperatura.

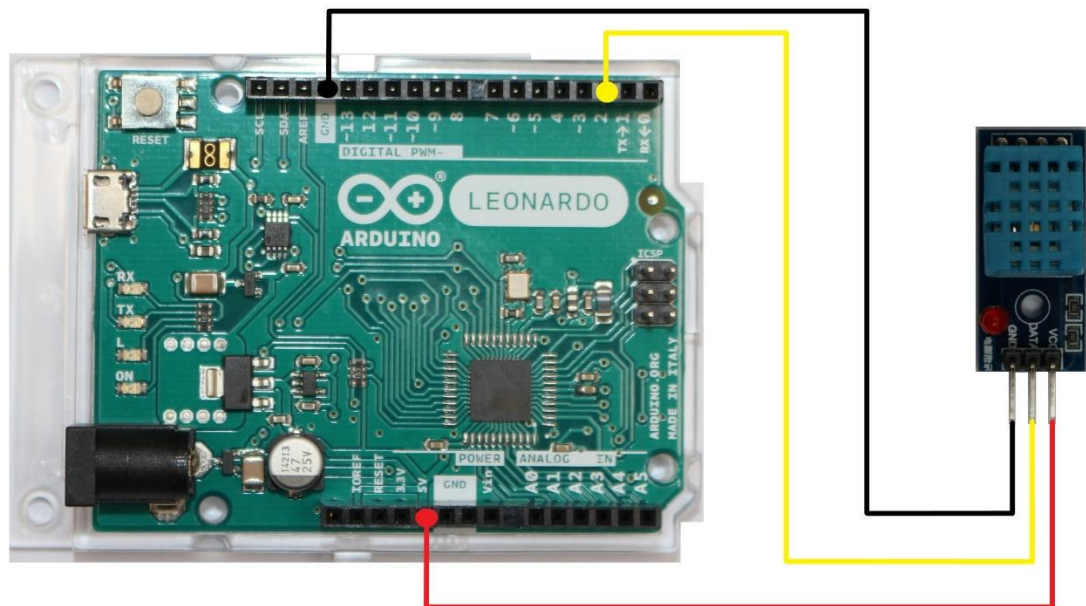
<u>0011 0101</u>	<u>0000 0000</u>	<u>0001 1000</u>	<u>0000 0000</u>	<u>0100 1001</u>
8 bits humedad	8 bits humedad	8 bits temperatura	8 bits temperatura	bits de paridad

El primer grupo de 8 bit es la parte entera de la humedad y el segundo grupo pertenece a la parte decimal. Lo mismo ocurre con el tercer y cuarto grupo, pero en este caso con la temperatura, que es el dato que nosotros extraemos.

El último grupo corresponde con los bits de paridad, para comprobar que no hay ningún bit erróneo. La comprobación es que al sumar los bits del primer al cuarto grupo tiene que ser igual a los bits de paridad. Como podemos ver en el ejemplo anterior.

La conexión de este sensor con Arduino es trivial como podemos ver en la Figura 33. Una vez en IDE de Arduino tenemos que añadir la librería 'Adafruit Unified Sensor' que como veremos en el guion de la práctica es algo sencillo.

Una vez descarga esta librería solo tendremos que hacer uso de ella e invocar el método `readTemperature()` que por defecto devuelve los datos en grados centígrados.



**Figura 33.** Conexión entre Arduino y sensor DHT11 (versión con PCB).

```

TEMPERATURA Arduino 1.8.12
Archivo Editar Programa Herramientas Ayuda

TEMPERATURA

#include "DHT.h"
#define dht11 DHT11
const int pinTemperatura = 2;
DHT dht(pinTemperatura, dht11);

void setup() {
  Serial.begin(9600);
  dht.begin();
}

void loop() {
  delay(2000);
  float temperatura = dht.readTemperature();
  Serial.print(temperatura);
  Serial.println(" °C");
}

```

4 Arduino Leonardo en COM5

**Figura 34.** Conexión entre Arduino y sensor DHT11 (versión con PCB).



## 5.5 Pulsador de 4 patas para Arduino

El pulsador dentro de nuestra práctica nos proporcionará la funcionalidad de **captura de datos**, es decir, al realizar una pulsación Arduino nos devolverá los datos del sensor en un momento concreto, **facilitando** al alumno la captura de los datos.

En nuestra práctica hemos implementado el pulsador con una **resistencia pull-down de 4K7  $\Omega$** , forzando a que la entrada lógica sea cero ('false') mientras esté en reposo.

La resistencia se conecta a tierra, de este modo cuando el pulsador este abierto la corriente se dirige hacia la resistencia dejando un valor de cero en el pin digital ('false') y cuando el pulsador se encuentra cerrado la corriente se moverá hacia el pin digital dejando un valor de uno ('true') como podemos observar en la Figura 35.

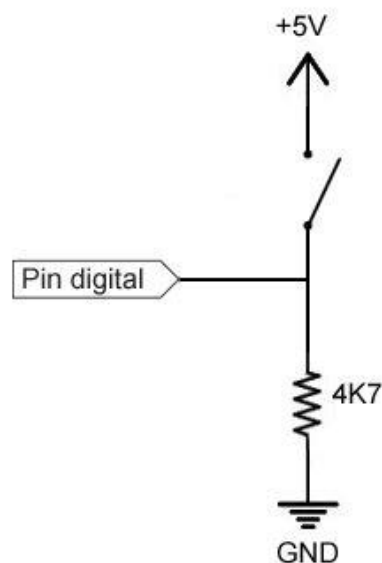


Figura 35. Esquema pulsador con resistencia pull-down.

En nuestra práctica la implementación de un pulsador para la toma de datos es clave debido a que el alumno puede realizar una captura de los datos del sensor, siendo **almacenados en un registro** como veremos en el capítulo 5.9, donde describiremos la aplicación para la recogida de datos de Arduino.

La implementación del pulsador y el resto de los elementos de la práctica son independientes entre sí. Para realizar una captura de datos Arduino se comprueba de forma recurrente el pin digital (4) para identificar si su valor es true "*high*" o false "*low*" como podemos ver en el código de la Figura 37. En el ejemplo de la Figura 37 podemos ver que el código es un ejemplo básico de funcionamiento del pulsador y para añadir la funcionalidad de captura de datos tendremos que añadir código adicional.

Este código se añadirá en la cláusula donde el valor del pin digital es high que corresponde cuando pulsamos el pulsador, como veremos en el guion de la práctica.

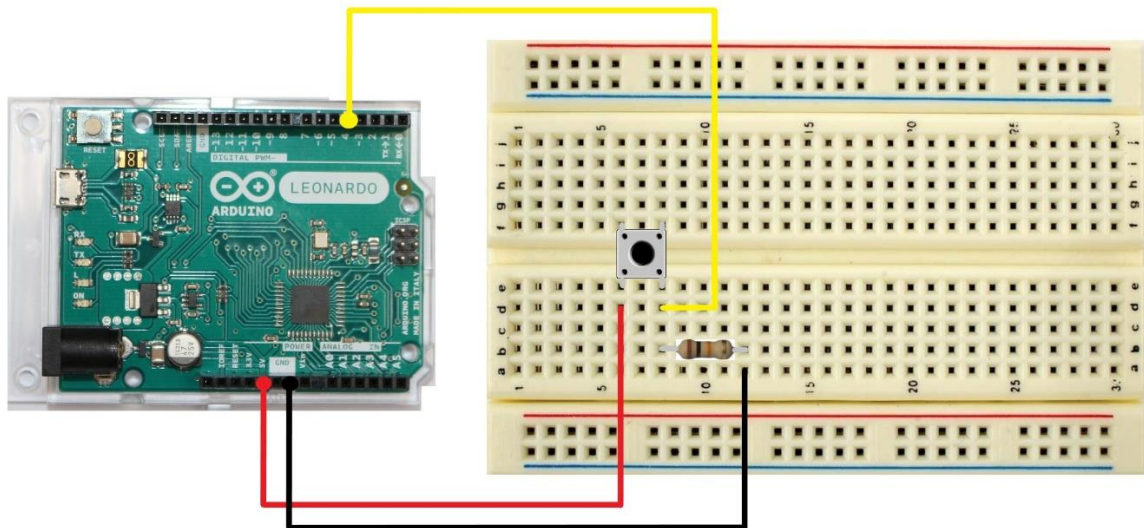


Figura 36. Esquema pulsador con resistencia pull-down.

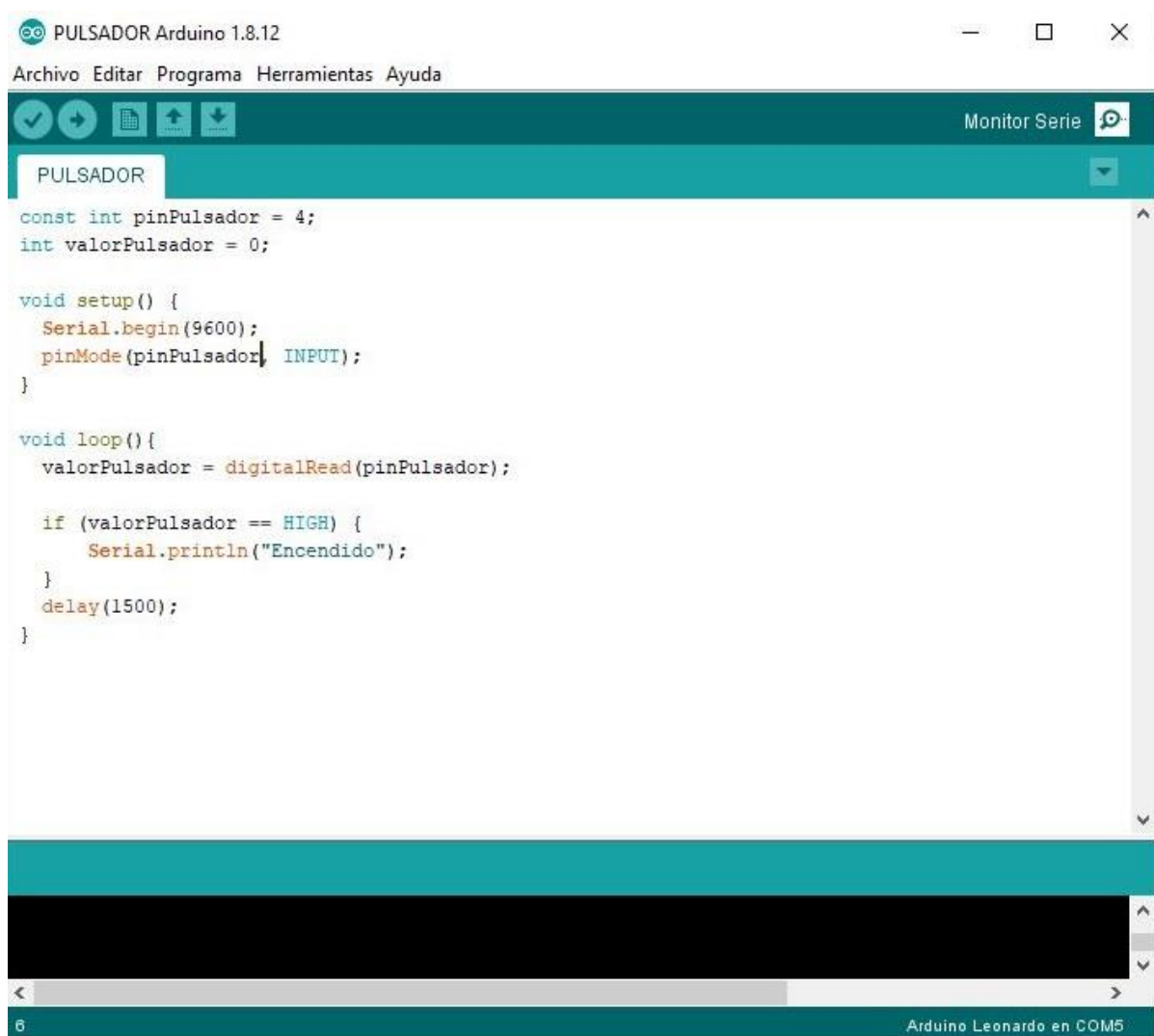


Figura 37. Conexión entre Arduino y pulsador 4 patas.



## 5.6 Medidor de radiación solar PCE-SPM 1

El medidor de radiación solar nos permitirá obtener datos de la **irradiancia** de nuestra fuente de luz, en nuestro caso unas **tiras LED**, las cuales nos centraremos en el apartado 4.7 y analizaremos el espectro de estas tiras LED.

El medidor reconoce la radiación solar completa, es decir, tanto la **radiación directa** (cielo despejado) como la **radiación solar difusa** (cielo nublado) del ambiente.

La mayor parte de la radiación solar que se refleja en la superficie de la tierra se encuentra en la **parte infrarroja visible** del espectro electromagnético, ya que la radiación ultravioleta es absorbida por la atmósfera, aunque el medidor sea capaz de medir la radiación en esta zona del espectro en nuestro caso sólo mediremos la parte del espectro visible debido a que nuestras tiras emiten radiación en la zona del espectro visible.



**Figura 38.** Medidor de radiación PCE-SPM 1.

Las principales **características técnicas** de este sensor son las siguientes:

- Rango de medición  $\rightarrow 0\text{-}2000 \text{ w/m}^2$
- Resolución  $\rightarrow 1 \text{ w/m}^2$
- Precisión  $\rightarrow \pm 10 \text{ w/m}^2$
- Rango espectral  $\rightarrow 400 \text{ a } 1100 \text{ nm}$
- Rango temperatura ambiental  $\rightarrow 0 \text{ a } 50 \text{ }^\circ\text{C}$

En nuestra práctica hemos situado el medidor PCE-SPM 1 a la misma altura en la cual colocaremos nuestra célula fotovoltaica para que las medidas de radiación por parte de nuestra fuente de luz sean lo más precisas posibles como podemos ver en la Figura 39. Como veremos el capítulo 4.6 tenemos un **soporte para colocar todos los elementos**, esta estructura está dividida por alturas, y tanto la célula fotovoltaica como el medidor de radiación se encuentran en la altura más baja del soporte.



**Figura 39.** Situación del medidor PCE-SPM-1 en el soporte.

## 5.7 Estructura para las mediciones

El soporte de nuestra práctica es uno de los elementos más importantes, ya que sirve tanto de soporte para nuestra fuente de luz, para asentar nuestra célula fotovoltaica y colocar los sensores de Arduino de una forma conjunta.

La intención de este soporte es tratar de **facilitar la toma de los datos de los sensores** de una forma unificada en cada alumno y que todas las capturas se realicen sin que se vean afectadas por **factores externos**.

Los principales **objetivos** a la hora de diseñar el soporte fueron las siguientes :

- **Toma de datos en alturas controladas.**

El alumno dispone de **cuatro alturas** para tomar las capturas de los sensores de Arduino por esa razón el soporte está diseñado **por partes**. Con esto conseguimos que la primera captura de datos se encuentre a una altura específica, y al quitar piezas del soporte esta altura disminuya de una forma uniforme hasta llegar a la última captura, permitiendo que el alumno tome  $n$  muestras dependiendo del número de piezas y con estas capturas dibuje la curva I-V de la célula fotovoltaica en el guion de la práctica.

- **Evitar ruido luminoso en las capturas de datos.**

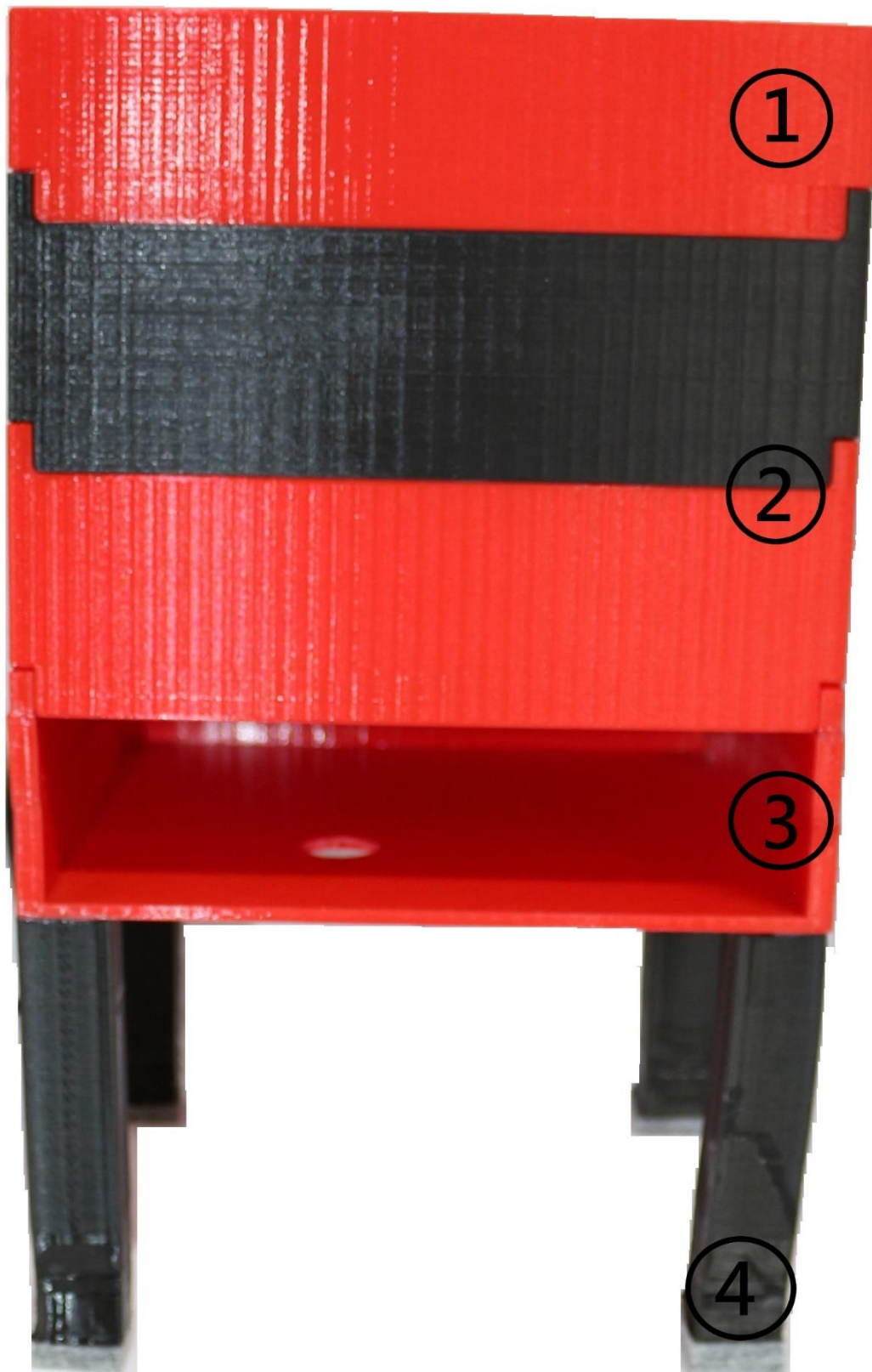
Un objetivo muy importante es intentar diseñar un soporte lo más **hermético** posible, es decir, que el ruido lumínico del exterior no entre en conflicto con nuestras medidas. Por esta razón **la forma del soporte es un cubo**, y tan solo está abierto en la parte baja del soporte por donde introduciremos nuestra célula fotovoltaica. Con esta estructura de cubo conseguimos **eliminar la mayor parte del ruido luminoso posible**.

- **Introducir al alumno en el mundo de las impresoras 3D.**

Algo también muy importante es que alumnos vean las **posibilidades** que ofrecen hoy en día las **impresoras 3D**. Con unos conocimientos muy básicos en diseño en 3D se pueden generar en unas horas una gran variedad de elementos. El objetivo es que alumno se interese por ampliar el conocimiento en esta área únicamente, durante la realización de la práctica **el alumno no diseñará ninguna pieza del soporte**.

- **Crear un espacio cerrado para aumentar la temperatura.**

Uno de los principales objetivos de la práctica es que los alumnos observen como **la temperatura afecta al comportamiento** de una célula fotovoltaica tal y como hemos visto en el capítulo 4.3, las primeras capturas de los datos del sensor se realizaran a temperatura ambiente. El segundo paso es **aumentar la temperatura** ambiente cercana a la célula fotovoltaica, por esta razón el diseño del soporte es el óptimo para aumentar la temperatura rápidamente debido a que es un **estructura cerrada y pequeña**.



**Figura 40.** Partes del soporte enumeradas.

La pieza (1) de nuestro soporte es **la parte superior**, por lo tanto, es la pieza elegida para albergar nuestro **sistema de iluminación** compuesto por dos tiras de leds como veremos en el siguiente capítulo 5.8.

Como podemos ver en la Figura 41, las tiras leds están colocadas rodeando la estructura de tal forma que a medida que bajamos la altura del soporte la célula pueda percibir más cantidad de luz. En caso de haber colocado las tiras de leds apuntando directamente sobre la célula fotovoltaica la diferencia de intensidad luminosa entre alturas habría sido mucho menor.



**Figura 41.** Parte superior del soporte.

Por último, cabe destacar que para intentar mantener el soporte lo más hermético posible he realizado **un pequeño corte** para pasar el cable de la fuente de alimentación al exterior.



Las piezas (2) son piezas cuya única función es la de **ampliar la altura entre la parte superior y la base del soporte**. El alumno tendrá que **quitar o añadir** estas piezas para realizar la captura de datos en distintas alturas siguiendo el guion de prácticas.

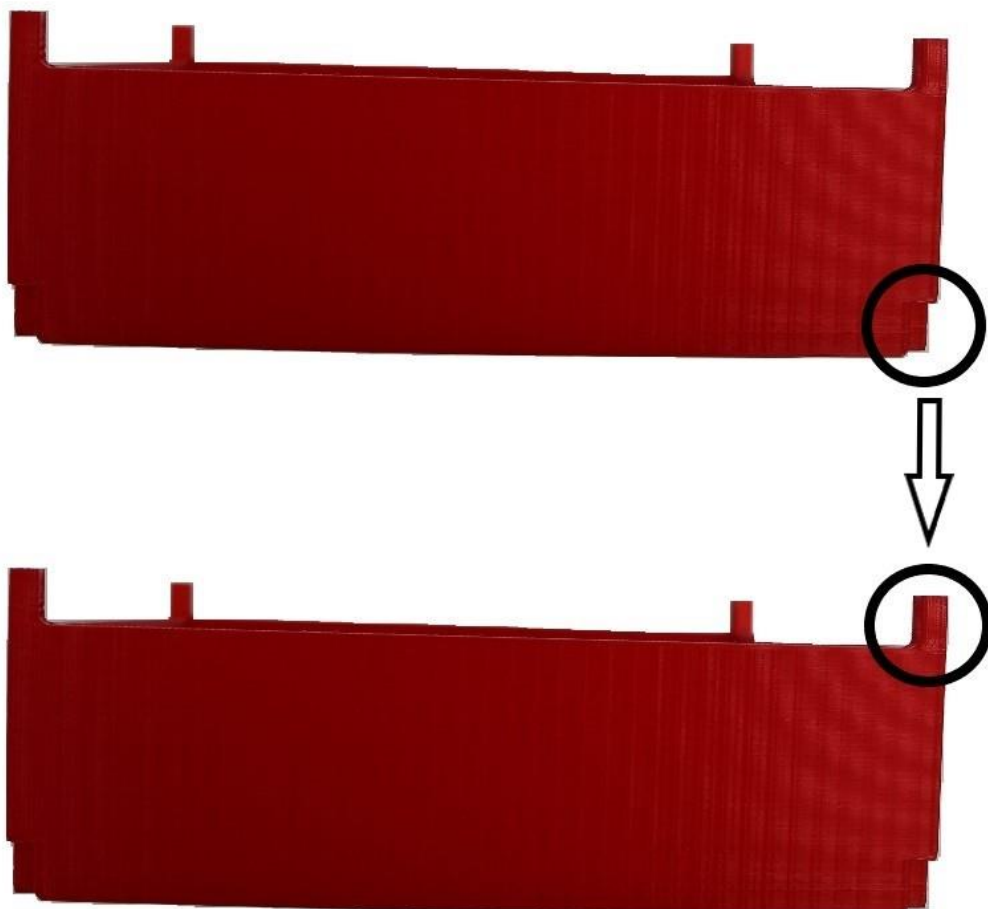
Cada pieza añade una **distancia de 5 cm**, que es la medida estándar de todas las piezas del soporte.

La pieza (3) es la más importante dentro de nuestro soporte, aquí se situará la célula fotovoltaica, sensor de temperatura, sensor irradiancia y, eventualmente, el sistema para calentar la célula.

El alumno tendrá que interactuar con esta pieza introduciendo la célula fotovoltaica por su apertura diseñada para la comodidad a la hora de **intercambiar las células fotovoltaicas** conforme al guion de prácticas.

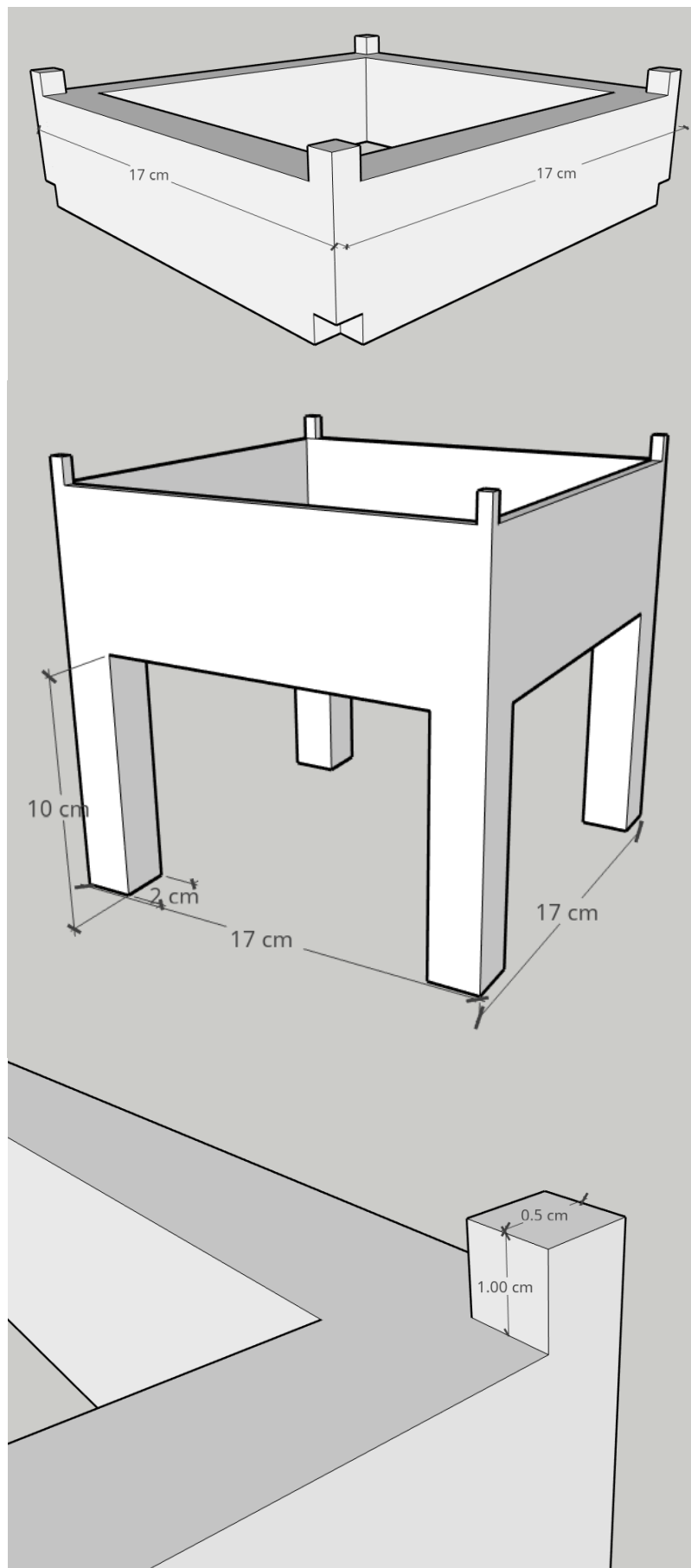
La pieza (4) son las “patas” de nuestro soporte, añadiendo la altura necesaria para que el sensor de irradiancia se encuentre a la misma altura de la célula fotovoltaica. El soporte se compone de 4 partes bien diferenciadas como podemos ver en la Figura 40.

Todas las piezas tienen una **parte sobresaliente** en forma de cubo en la parte superior de cada pieza y en su parte inferior hay un hueco con forma de cubo con las mismas dimensiones, pudiendo así **encajar** todas las piezas entre si de una manera **sencilla y rápida** como podemos ver en la siguiente Figura 42.



**Figura 42.** Fijación entre piezas del soporte.





**Figura 43.** Dimensiones del soporte.

## 5.8 *Tiras LED con fuente de alimentación*

En nuestra práctica vamos a analizar el comportamiento de las células fotovoltaicas dependiendo de factores como la temperatura, cantidad de radiación o cercanía a la fuente luminosa. Lo más relista posible habría sido construir un soporte con un **simulador solar** pero al tratarse de una práctica académica nos hemos visto obligados a tener que usar una fuente de luz más **económica** para nuestro propósito.

En nuestro caso hemos elegido **tiras de led de luz blanca** para nuestra práctica, en concreto hemos conectado 2 tiras led en paralelo como podemos ver en la Figura 41 con una fuente de alimentación de 12V, 2A.

Para evitar que siempre obtengamos mucha radiación sobre la célula fotovoltaica, hemos colocado la tira por dentro de la parte superior, y una parte muy pequeña apuntando a la célula fotovoltaica como podemos ver en la Figura 41.

## 5.9 *Aplicación conexión con Arduino*

Para facilitar la captura de datos al alumno nos hemos decidido por implementar una pequeña **aplicación open-source** desarrollada por “panama hitek” y cuyo nombre es ArduinoDataLogger versión 2. Está desarrollada en java y para facilitar aún más el trabajo a los alumnos hemos creado un **archivo ejecutable** de la aplicación para la conexión con Arduino.

El funcionamiento de la aplicación es muy intuitivo, primero tenemos que añadir los sensores que tiene conectados nuestra placa Arduino, podemos darle nombre y unidades en este proceso. Después, la aplicación se conecta al puerto serie de Arduino para recolectar todos los datos de los sensores y los presenta en una tabla. Además, una vez terminada la captura de datos, la aplicación nos permite **exportar los datos a Excel** para completar el guion de la práctica. La **página principal** de la aplicación podemos verla en la figura 43, siendo esta la única modificación del código fuente, añadiendo y personalizando la página principal de la aplicación.



## 6. Guion de la práctica

# Introducción a la caracterización de Células Fotovoltaicas mediante Arduino

---



### 1. Objetivos

Esta práctica tiene como objetivo principal iniciarse en el hardware open source con Arduino y la obtención de las curvas características de algunos de los más importantes tipos de células fotovoltaicas que encontramos en el mercado.

También cabe destacar la importancia del uso de impresoras en 3D para diseñar piezas a medidas de una manera económica, como veremos en el soporte realizado para esta práctica.

### 2. Introducción

En el mercado actual podemos encontrar multitud de hardware open source, en esta práctica usaremos Arduino. Este hardware está diseñado para un ámbito académico desde su inicio, aunque también es usado en el entorno profesional gracias a todos los beneficios que aportan estos sistemas.

Para caracterizar las células fotovoltaicas vamos a emplear varios sensores conectados a la placa Arduino para obtener la corriente, el voltaje y la temperatura ambiente. Arduino permite la implementación de una manera muy sencilla y práctica como veremos en los siguientes apartados. Para capturar los datos conectaremos nuestro PC a la placa Arduino a través del puerto USB, suministrando la energía por este puerto, evitando el uso de fuentes de alimentación externas.

### 3. Configuración IDE Arduino

Arduino es una placa basada en un microcontrolador ATMEL. Estos microcontroladores son circuitos integrados en los que se pueden grabar instrucciones, las cuales se escriben en el entorno IDE de Arduino.

El IDE es una aplicación multiplataforma para Windows, macOS y Linux que está diseñada en Java. También es posible acceder al IDE web de Arduino.

Para instalar del IDE de Arduino debemos seguir los siguientes pasos:

1. En un navegador web copiamos la siguiente dirección  
<https://www.arduino.cc/en/main/software>
2. Seleccionamos nuestro **sistema operativo**, en nuestro caso Windows.



Figura 1. Sitio web para la descarga del IDE de Arduino.

3. Una vez descargado el fichero zip, **descomprimos el fichero y ejecutamos el siguiente fichero** para abrir el IDE de Arduino.

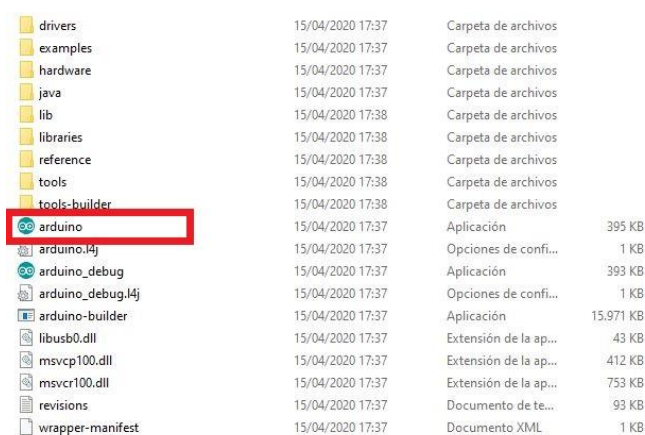
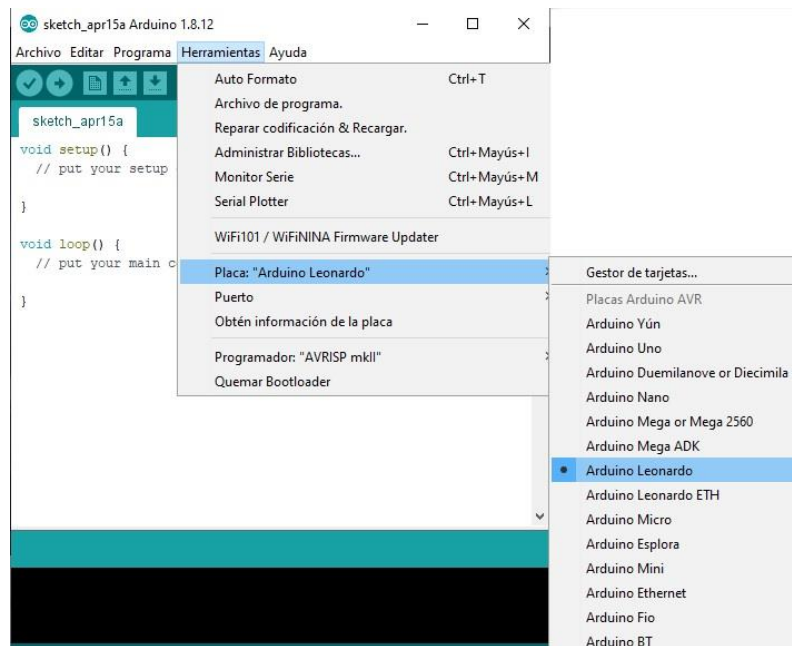


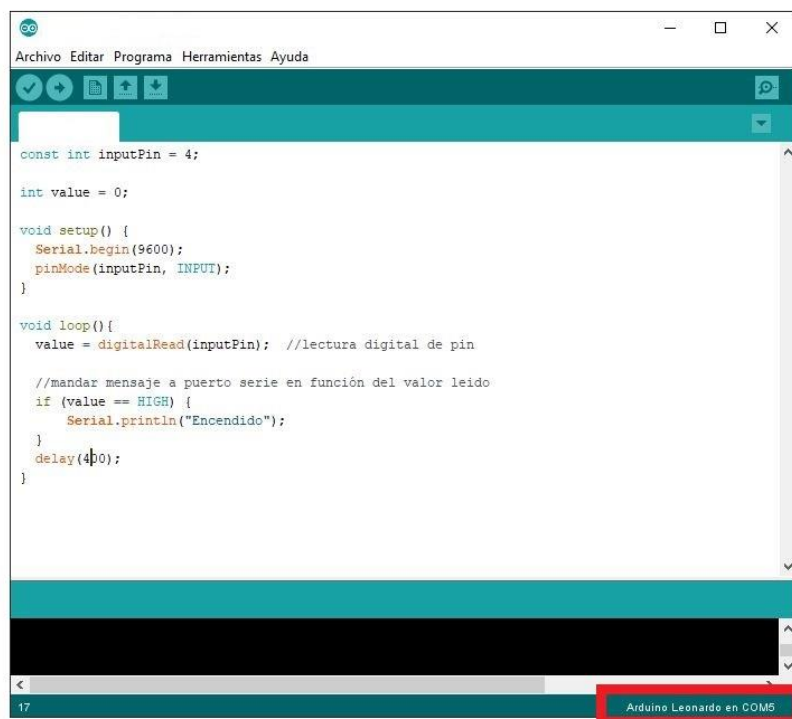
Figura 2. Fichero ejecutable Arduino.

4. Con el IDE una vez abierto, debemos **configurar** nuestra placa Arduino para poder cargar ficheros. Para ello, seguimos la siguiente ruta **Herramientas/Placa** y debemos seleccionar nuestra placa, **Arduino Leonardo en nuestro caso**.



**Figura 3.** Configuración placa Arduino en el IDE.

5. Por último, debemos conectar nuestra placa Arduino por el puerto USB al PC y seguir la siguiente ruta **Herramientas/Puerto** y seleccionar el puerto USB que el PC ha asignado a la placa Arduino. Una vez realizado estos pasos podemos comprobar que la conexión entre Arduino y el PC se realizó con éxito observando lo siguiente:



**Figura 4.** IDE de Arduino correctamente configurado.

6. Para comprobar si nuestro código Arduino tiene algún error debemos hacer clic sobre el **botón de color amarillo**. Para subir el código a nuestra placa Arduino debemos pulsar el **botón de color rojo**. Si el fichero se sube correctamente escucharemos un aviso sonoro de nuestro PC.



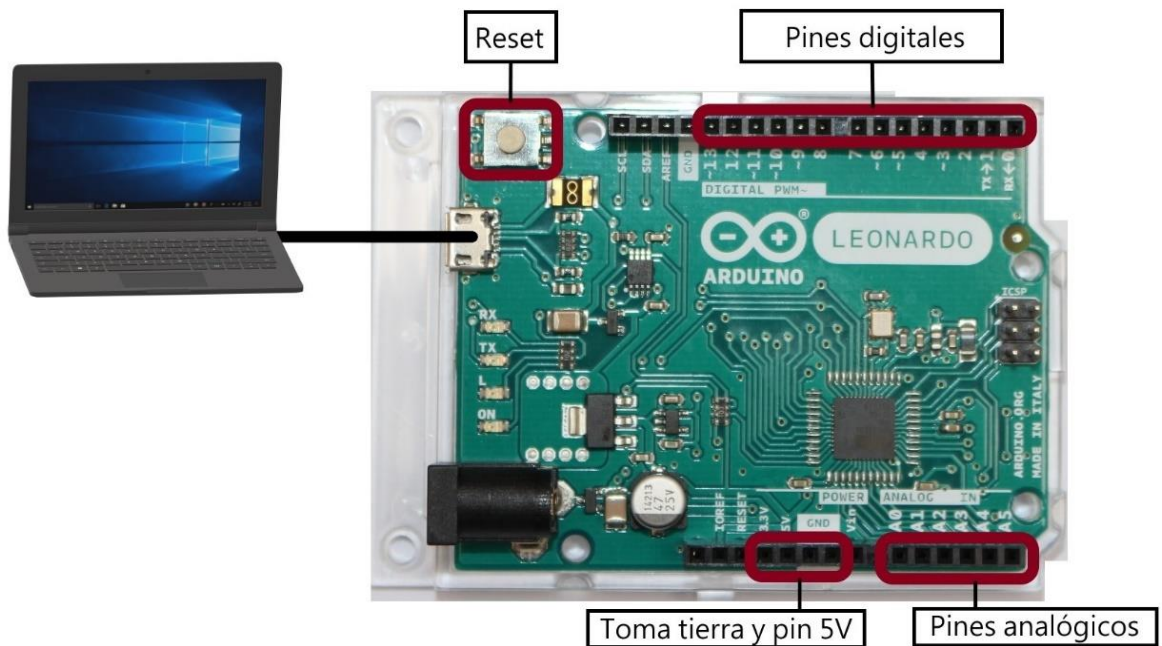
**Figura 5.** Botones de verificación y subida de ficheros Arduino.

7. Arduino devuelve los datos a través del monitor serie, para acceder pulsamos el botón **de color azul**.



**Figura 6.** botón para ejecutar el monitor serie de Arduino.

La placa Arduino Leonardo tiene varias **entradas digitales y analógicas**, además permite realizar un **reset** de la placa en aquellas ocasiones que se puede quedarse bloqueada como podemos ver en la siguiente imagen.



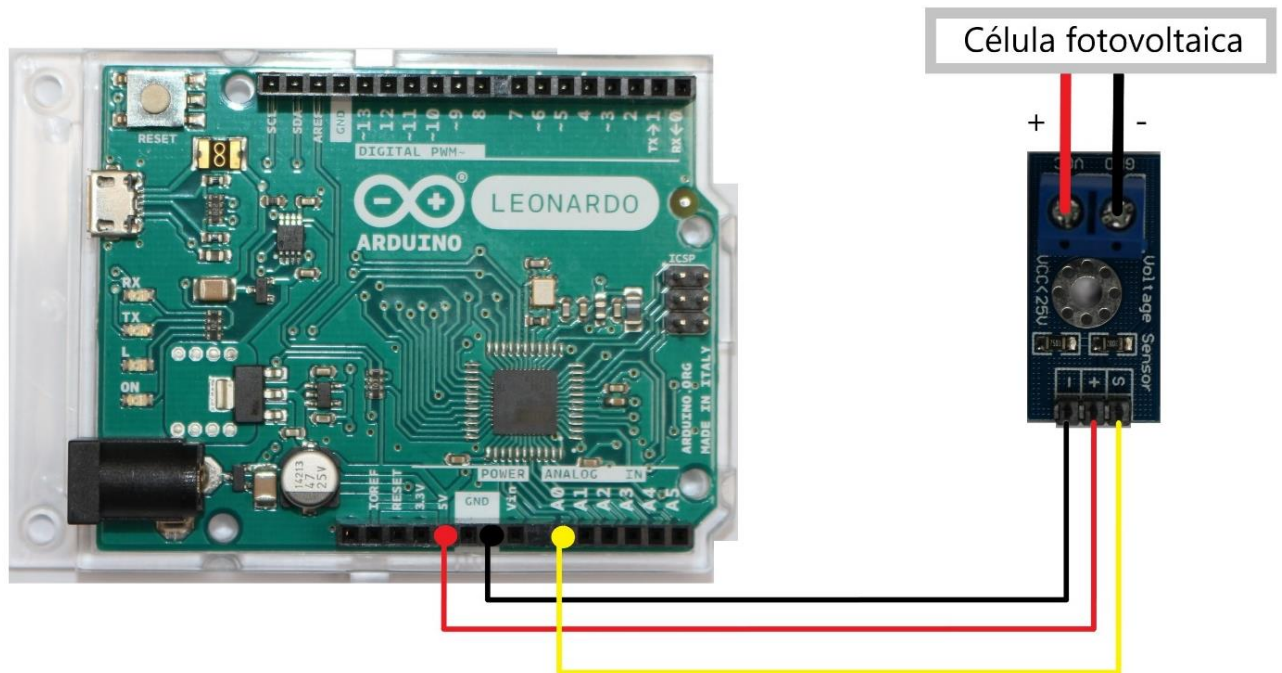
**Figura 7.** Esquema pines Arduino.



Una vez configurada nuestra placa Arduino, es hora de empezar a **conectar sensores** y poder obtener los datos de ellos. En la práctica vamos a capturar los datos de **tres sensores simultáneamente**, pero para empezar a adquirir conceptos sobre Arduino vamos a realizar una configuración independiente de cada sensor.

### 3.1 Sensor FZ0430

El sensor FZ0430 es el encargado de recoger datos del **voltaje**.



**Figura 8.** Esquema conexión entre Arduino y sensor FZ0430.

1. Interconectamos Arduino y nuestro sensor de la siguiente manera:

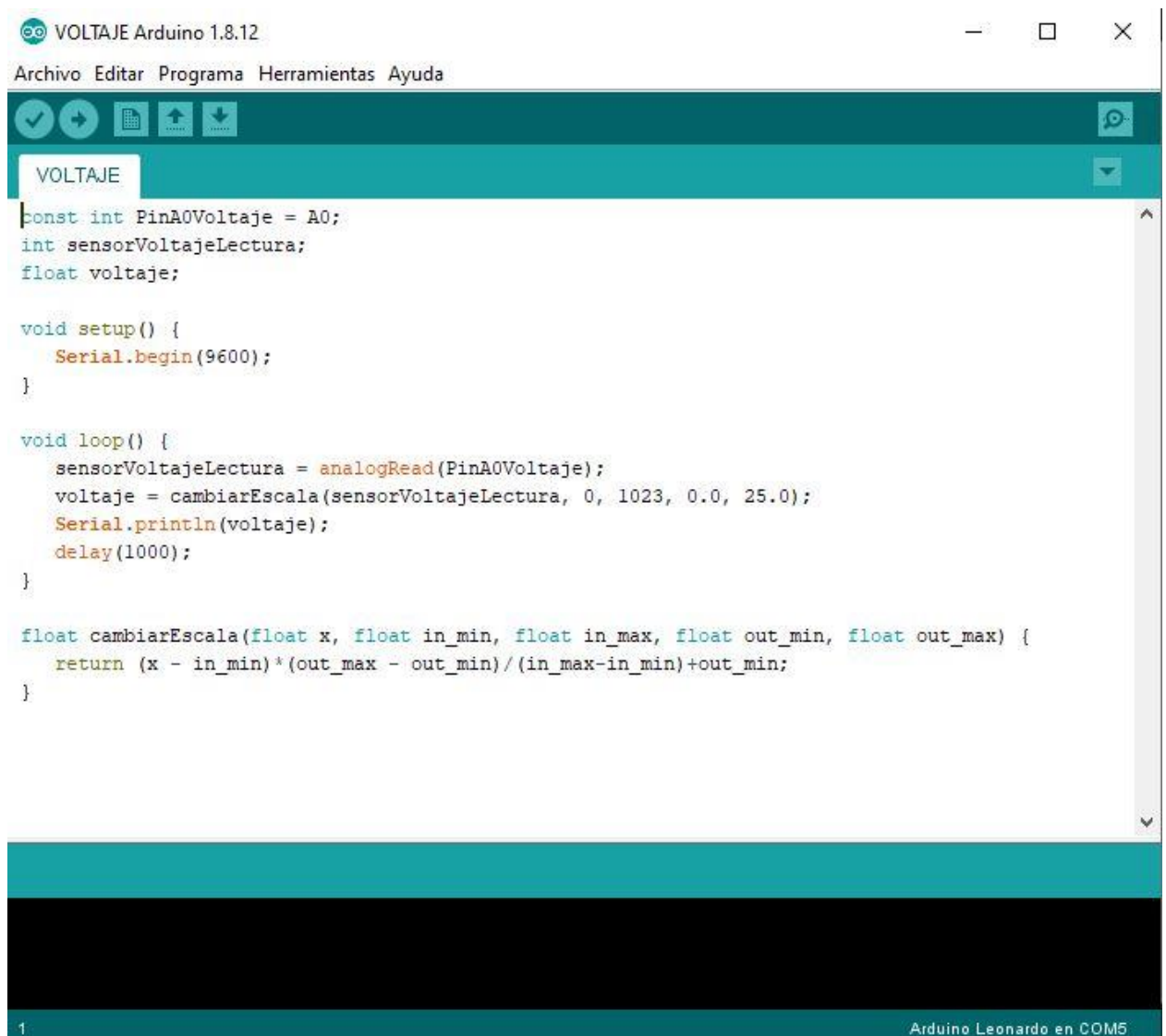
- S (sensor) → Pin analógico (Arduino)
- + (sensor) → Pin de 5V (Arduino)
- (sensor) → Pin GND (Arduino)

Es importante conectarlo a **los pines analógicos** de la placa Arduino, ya que el sensor devuelve datos analógicos.

2. Interconectamos sensor con la célula fotovoltaica o una fuente de alimentación.

- VCC (sensor) → Polo positivo (fuente de alimentación)
- GND (sensor) → Polo negativo (fuente de alimentación)

3. En el IDE de Arduino tenemos que añadir el siguiente código:



```

const int PinA0Voltaje = A0;
int sensorVoltajeLectura;
float voltaje;

void setup() {
  Serial.begin(9600);
}

void loop() {
  sensorVoltajeLectura = analogRead(PinA0Voltaje);
  voltaje = cambiarEscala(sensorVoltajeLectura, 0, 1023, 0.0, 25.0);
  Serial.println(voltaje);
  delay(1000);
}

float cambiarEscala(float x, float in_min, float in_max, float out_min, float out_max) {
  return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;
}

```

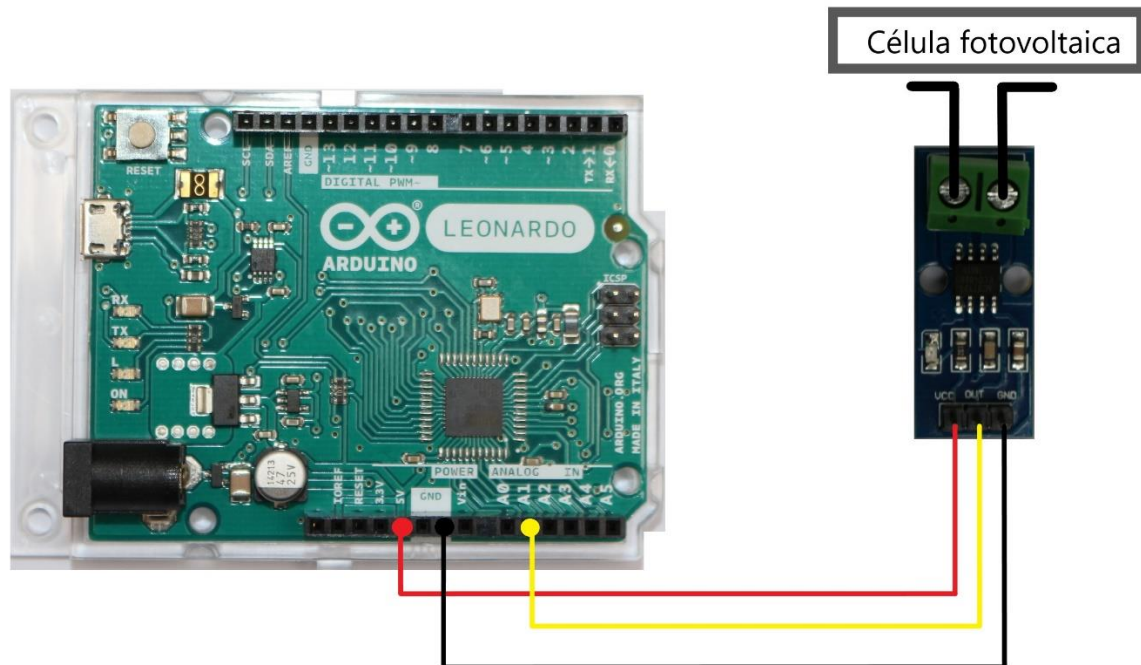
**Figura 9.** Código Arduino sensor FZ0430.

Es importante tener en cuenta el pin usado para conectar el sensor porque como podemos ver en la Figura 3, instanciamos una constante con el pin usado en nuestro caso **A0**. Para leer los datos del sensor usamos la función *analogRead()* que está incluida en la librería de Arduino. Para pintar el resultado a través del monitor serie usamos el método estático `println("lectura del sensor")` de la clase `Serial`.

4. Subimos el fichero a la placa y abrimos el monitor serie para ver los datos capturados por el sensor (ver Figuras 5 y 6).

### 3.2 Sensor ACS712

El sensor ACS712 es el encargado de recoger datos de la **corriente**.



**Figura 10.** Esquema conexión entre Arduino y sensor ACS712.

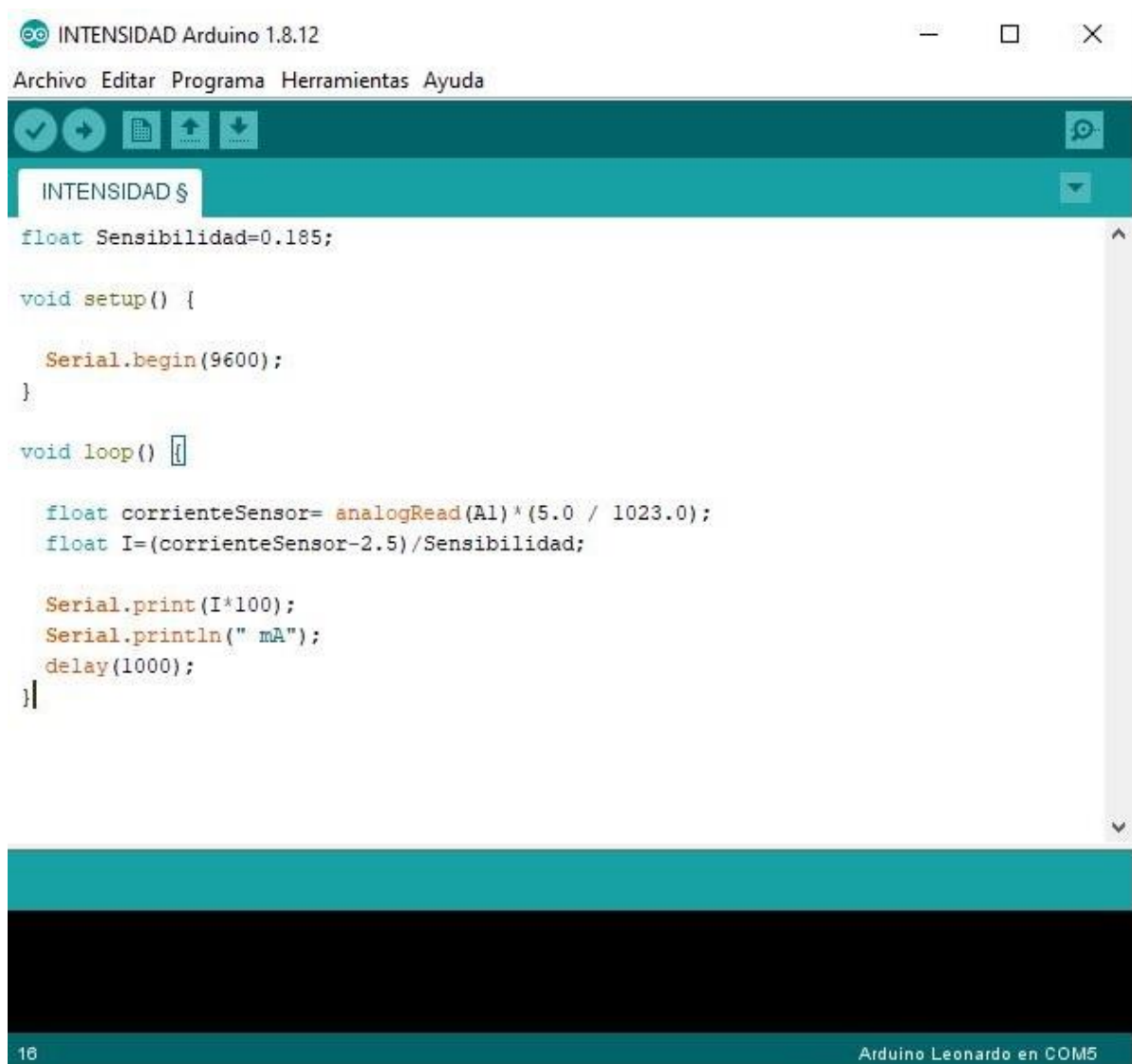
1. Interconectamos Arduino y nuestro sensor de la siguiente manera:

- OUT** (sensor) → Pin analógico (Arduino)
- VCC** (sensor) → Pin de 5V (Arduino)
- GND** (sensor) → Pin GND (Arduino)

Es importante conectarlo a **los pines analógicos** de la placa Arduino, ya que el sensor devuelve datos analógicos.

2. Interconectamos sensor con la célula fotovoltaica o una fuente de alimentación de forma que podamos medir la corriente que pasa a través del sensor. Como podemos ver en la Figura 10, para medir corriente, el sensor se debe conectar en una zona donde este el **circuito abierto** y el sensor cierra el circuito y mide la corriente que lo atraviesa.

3. En el IDE de Arduino tenemos que añadir el siguiente código:



```

INTENSIDAD Arduino 1.8.12
Archivo Editar Programa Herramientas Ayuda

float Sensibilidad=0.185;

void setup() {
  Serial.begin(9600);
}

void loop() {
  float corrienteSensor= analogRead(A1) * (5.0 / 1023.0);
  float I=(corrienteSensor-2.5)/Sensibilidad;

  Serial.print(I*100);
  Serial.println(" mA");
  delay(1000);
}
  
```

16 Arduino Leonardo en COM5

**Figura 11.** Código Arduino sensor ACS712.

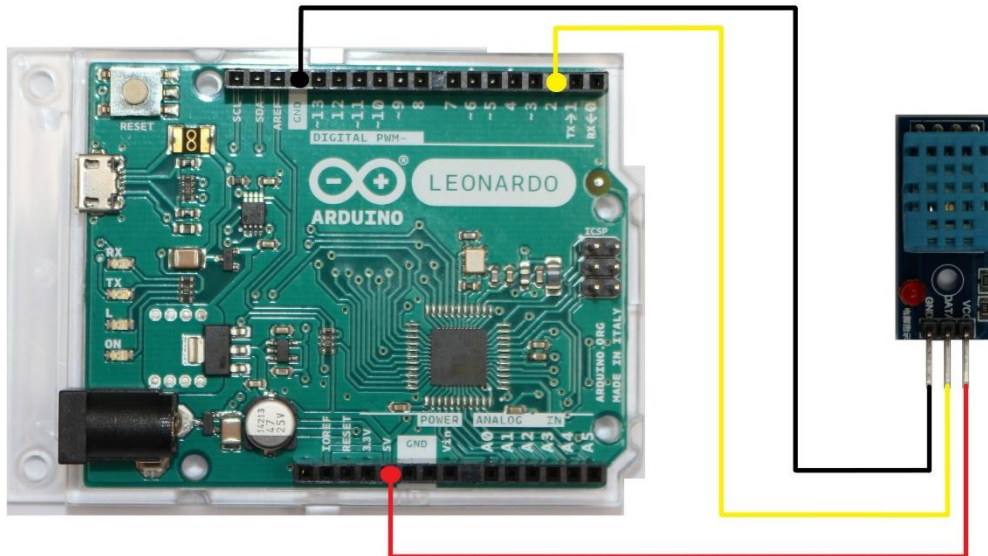
Es importante tener en cuenta el pin usado para conectar el sensor porque como podemos ver en la Figura 11, para realizar una lectura del sensor debemos llamar al método ***analogRead(A1)*** siendo este el parámetro que hemos mencionado. Para calcular la corriente debemos implementar la siguiente ecuación, en donde la variable V hace referencia al nivel de voltaje medida por el sensor, en el código se corresponde con la variable “corrienteSesnor”.

$$I = \frac{V - 2,5}{m}$$

4. Subimos el fichero a la placa y abrimos el monitor serie para ver los datos capturados por el sensor (ver Figuras 5 y 6).

### 3.3 Sensor DHT11

El sensor DHT11 es el encargado de medir la **temperatura ambiente** y su posterior transformación en datos digitales para enviarlo después a la placa Arduino.



**Figura 12.** Esquema conexión entre Arduino y sensor DHT11.

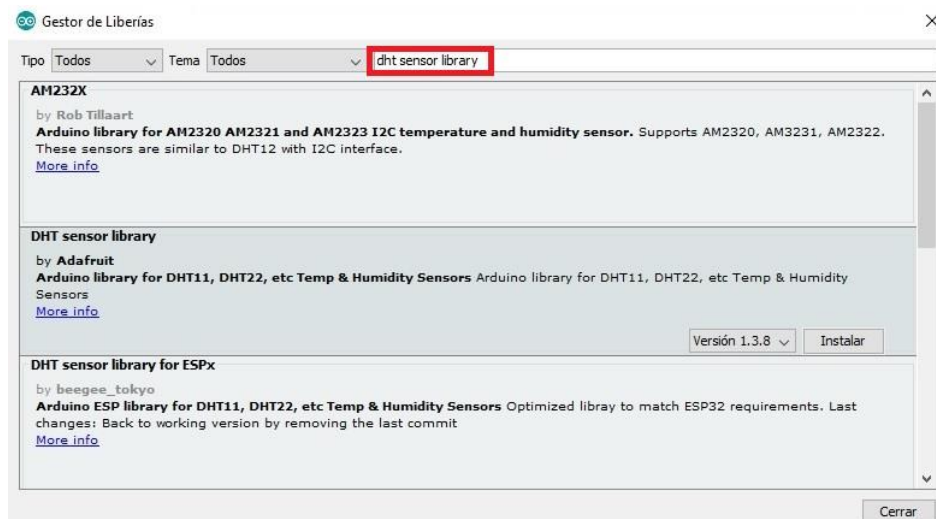
1. Interconectamos Arduino y nuestro sensor de la siguiente manera:

**DATA** (sensor) → Pin digital (Arduino)  
**VCC** (sensor) → Pin de 5V (Arduino)  
**GND** (sensor) → Pin GND (Arduino)

El sensor de temperatura **debe situarse lo más cerca posible de la célula fotovoltaica** para que los datos de temperatura ambiente sean lo más precisos posible.

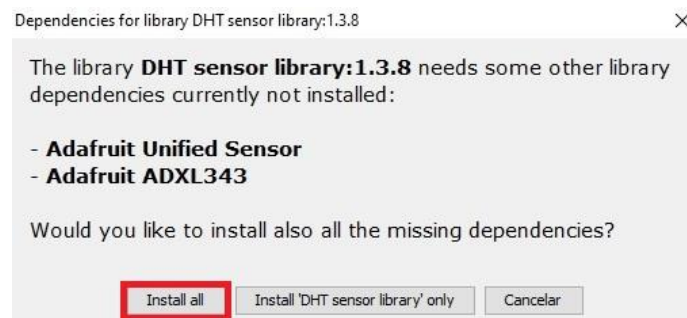
2. Añadir librerías Arduino para el sensor:

Para añadir librerías debemos abrir el gestor de librerías para ello siga la siguiente ruta **Herramientas/Administrar Bibliotecas**, una vez abierto el gestor, buscamos:

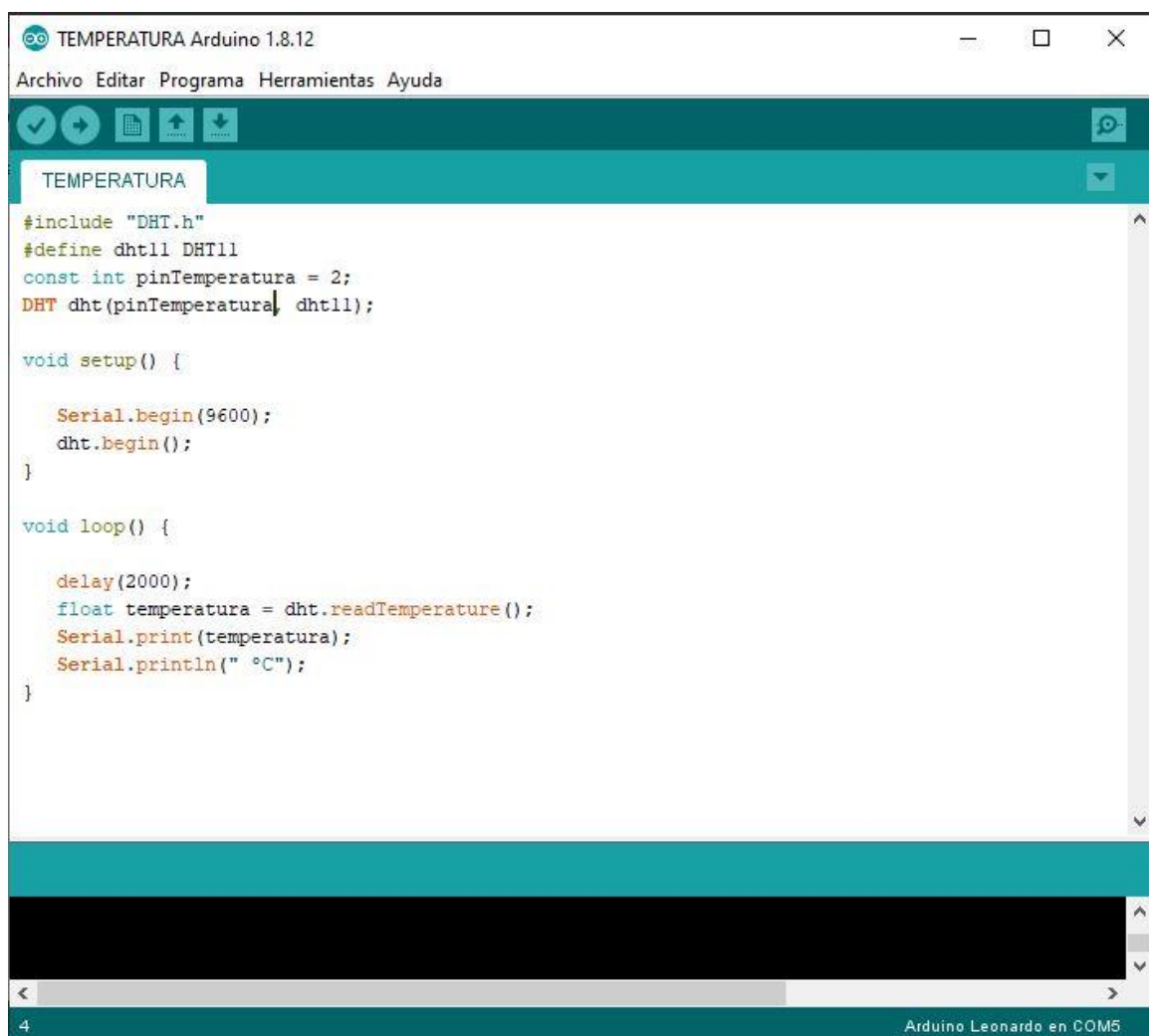




Por último, **instalamos** todas las dependencias de Arduino.



3. En el IDE de Arduino tenemos que añadir el siguiente código:



**Figura 13.** Código Arduino sensor DHT11.

Es importante tener en cuenta que el sensor devuelve los datos en formato **digital**. Para leer los datos del sensor solo tenemos que hacer uso del método *readTemperature()* de la clase DHT, esta clase pertenece a la librería importada en el paso anterior.

4. Subimos el fichero a la placa y abrimos el monitor serie para ver los datos capturados por el sensor (ver Figuras 5 y 6).



### 3.4 Pulsador 4 patas

En nuestra práctica hemos implementado el pulsador con **una resistencia pull-down** de  $4K7\ \Omega$ , forzando a que la entrada lógica sea cero ('false') mientras esté en reposo. La resistencia se conecta a tierra, de este modo cuando el pulsador este abierto la corriente se dirige hacia la resistencia dejando un valor de cero en el pin digital ('false') y cuando el pulsador se encuentra cerrado la corriente se moverá hacia el pin digital dejando un valor de uno ('true').

Para realizar una captura de datos Arduino **se comprueba de forma de recurrente el pin digital** usado para identificar si su valor es true "high" o false "low". En el ejemplo de la Figura 14 podemos ver que el código es un **ejemplo básico de funcionamiento** del pulsador y para añadir la funcionalidad de captura de datos de los sensores debemos añadir código dentro de la cláusula if, esta parte del código solo se ejecutará cuando presionemos este pulsador.

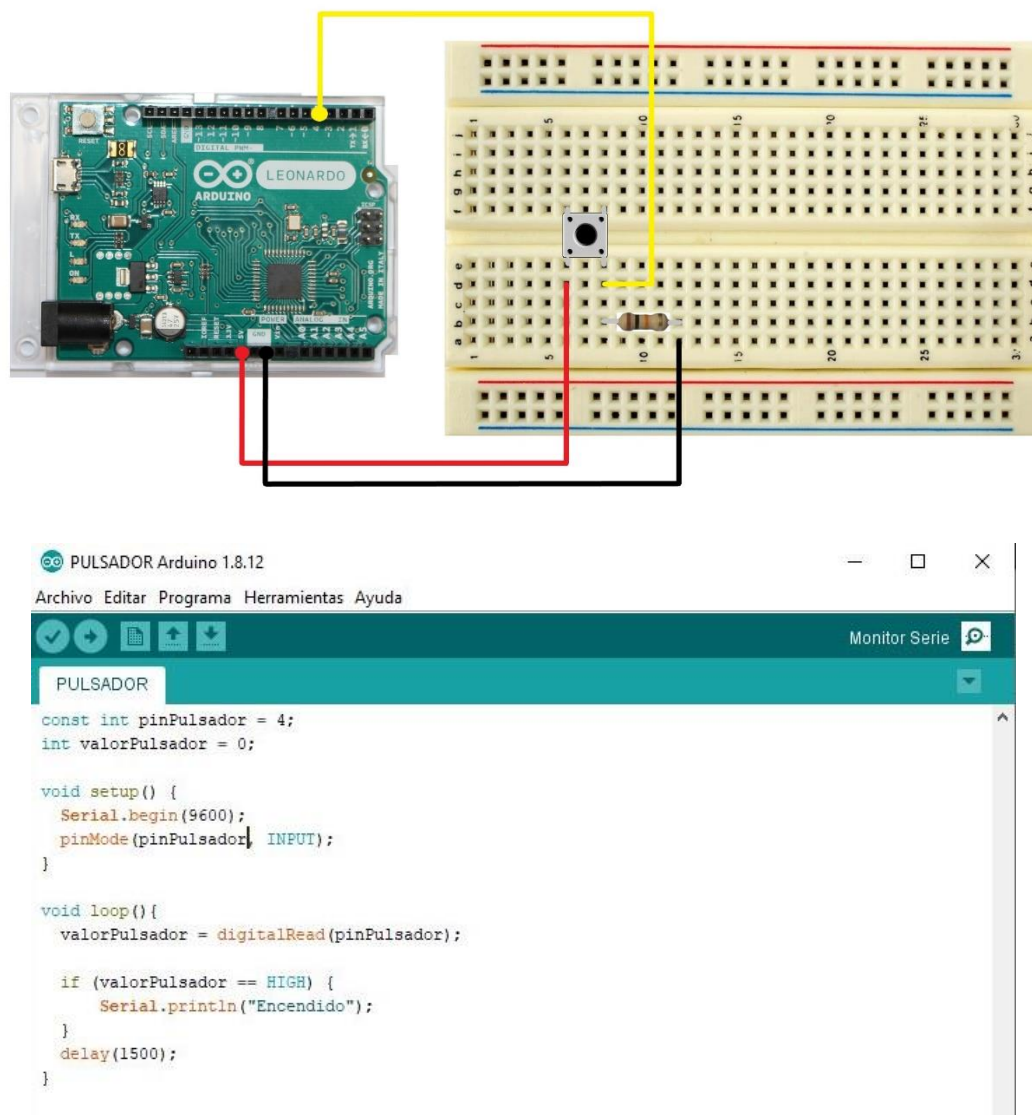
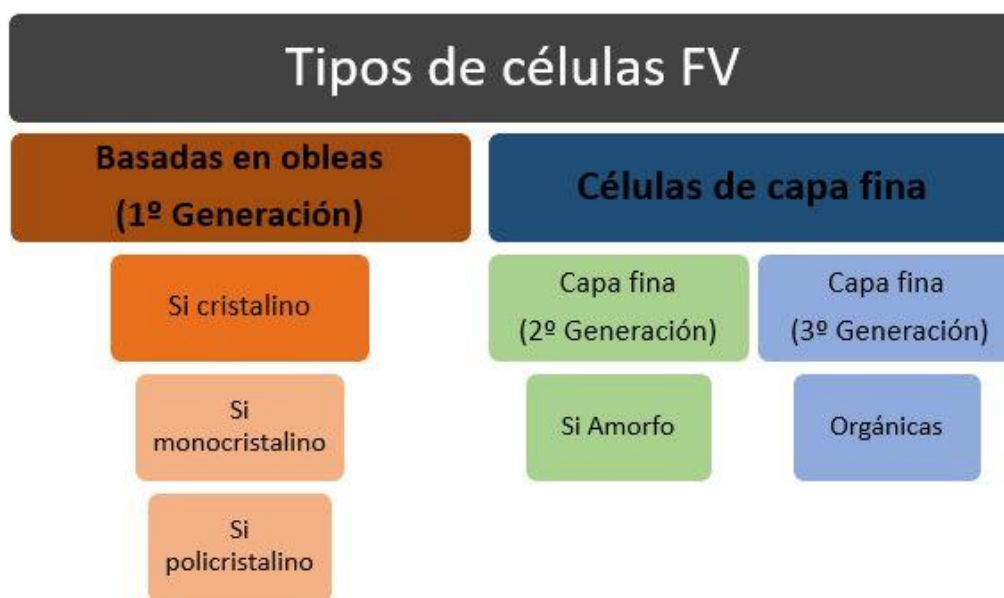


Figura 14. Esquema conexión y código del pulsador 4 patas.

## 4. Introducción Células Fotovoltaicas

En la práctica vamos a analizar 4 células, caracterizando la curva I-V con distintos niveles de irradiancia y a distintas temperaturas. Para ello, vamos a ver las diferencias entre las distintas tecnologías de estas células fotovoltaicas.



**Figura 15.** Tipos de células usadas en la práctica.

El Si Cristalino se corresponde con la denominada **primera generación**, siendo la tecnología predominante a nivel comercial con más de un 90% del mercado mundial de células fotovoltaicas. Dentro de la categoría de Si cristalino encontramos el **Si monocristalino** que es de una calidad ligeramente superior al **Si policristalino**, así como su precio de producción. Con relación a su eficiencia, el Si monocristalino se encuentra de media en un 20% mientras que el Si policristalino se sitúa entre un 16%. Esto se traduce en que el Si monocristalino parece superior en todos los sentidos, pero ¿cuáles serán sus cuotas de mercado? el Si monocristalino tiene un 35% y el Si policristalino un 55%. Esto es debido a que, aunque el Si monocristalino es mejor, su precio de producción es también superior.

El Si amorfo se encuadra en la **segunda generación** de capa fina, cuyo nombre se debe a que el espesor de estas células fotovoltaicas es mucho mejor que las de primera generación, pero también se traduce en una eficiencia inferior en torno a un 6-8%. Su penetración en el mercado se reduce a un 2% aunque seguramente estés usando esta tecnología en este momento ya que se encuentran en calculadoras alimentadas por células fotovoltaicas.

Las células fotovoltaicas orgánicas se encuadran dentro de la **tercera generación**, es un tipo de célula fotovoltaica en la que se emplea electrónica orgánica para generar corriente eléctrica a través del efecto fotoeléctrico. Esta tecnología se basa en el uso de materiales polímeros orgánicos conductores o pequeñas moléculas orgánicas. La principal ventaja es que permite una producción a gran escala con unos costes muy reducidos. Sin embargo, tiene una eficiencia y una estabilidad escasa en comparación las de Si cristalino, aunque se prevé que mejore en los próximos años.

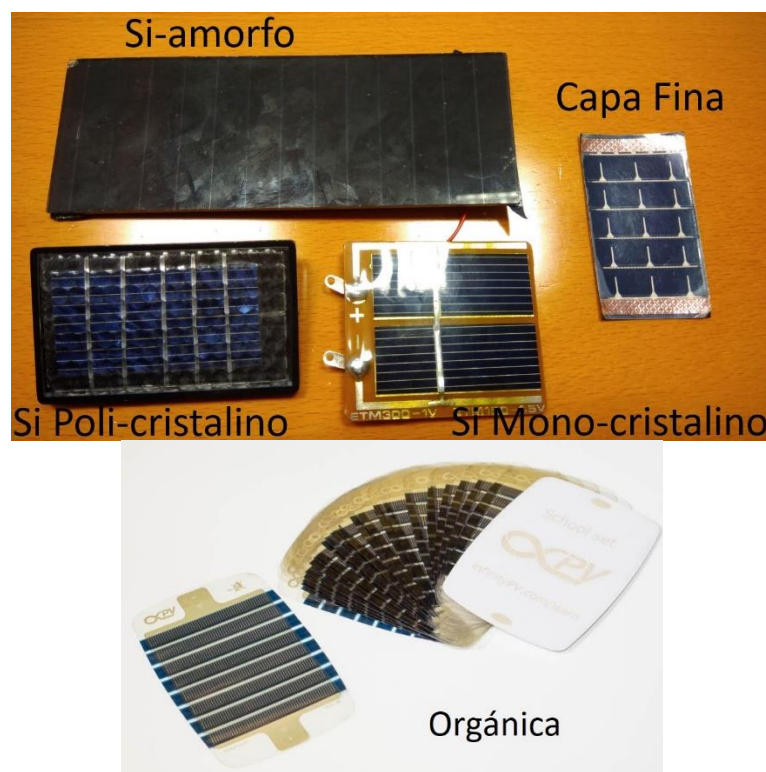
Por último, es importante tener una visión general de las **diferencias entre las distintas tecnologías** usadas en la práctica, tales como la eficiencia, estabilidad y el coste de su producción como podemos observar en la siguiente tabla.

Tipo de célula	Eficiencia	Estabilidad	Coste producción
<b>Si monocristalino</b>	Muy alta	Excelente	Muy alto
<b>Si policristalino</b>	Muy alta	Excelente	Medio
<b>Capa fina (Si amorfo)</b>	Baja	Muy baja	Bajo
<b>Célula orgánica</b>	Baja	Baja	Muy bajo

**Figura 16.** Comparación de las tecnologías en células fotovoltaicas.

Al margen de estos conceptos, es importante resaltar el **impacto medioambiental** de cada tecnología. Algunas tecnologías utilizan compuestos que pueden ser peligrosos para el medioambiente a la hora de ser retirados y reciclados. Las células de capa fina son las potencialmente más contaminantes de estas tecnologías, dejando a las células orgánicas como las menos contaminantes con una gran diferencia respecto a las demás.

El principal objetivo de una tecnología es intentar conseguir eficiencias superiores con los menores costes de producción.



**Figura 17.** Células empleadas en la práctica.

## 5. Desarrollo Práctica

El objetivo de la práctica es **caracterizar los cuatro tipos de tecnologías** de células fotovoltaicas descritas anteriormente, apoyándonos en la facilidad que aporta Arduino para conectar y capturar datos de los sensores. El proceso de captura de datos se llevará a cabo mediante distintas medidas con la fuente de luz, motivo por el que se ha diseñado un soporte fabricado con la impresora 3D del grupo encargado de impartir esta asignatura.

### 5.1 Soporte

El soporte tiene **cuatro piezas diferentes** como podemos apreciar en la Figura 18, en la pieza '1' se encuentra la fuente de luz compuesta por dos tiras de Leds conectadas en paralelo a una fuente de alimentación. La función de la pieza '2' es únicamente la de añadir o disminuir distancia entre la fuente de luz y la célula fotovoltaica, cabe destacar que las piezas tienen todas una altura estándar de 5 cm. La pieza '3' sirve para introducir la célula fotovoltaica que tengamos que caracterizar además del sensor de temperatura. Por último, la pieza '4' son las patas de nuestro soporte y tienen el tamaño justo para poner adaptar el sensor de irradiancia como hemos visto en el capítulo 3.4.



**Figura 18.** Partes del soporte diseñado e implementado mediante el uso de una impresora 3D.



## 5.2 Conexión entre sensores y Arduino

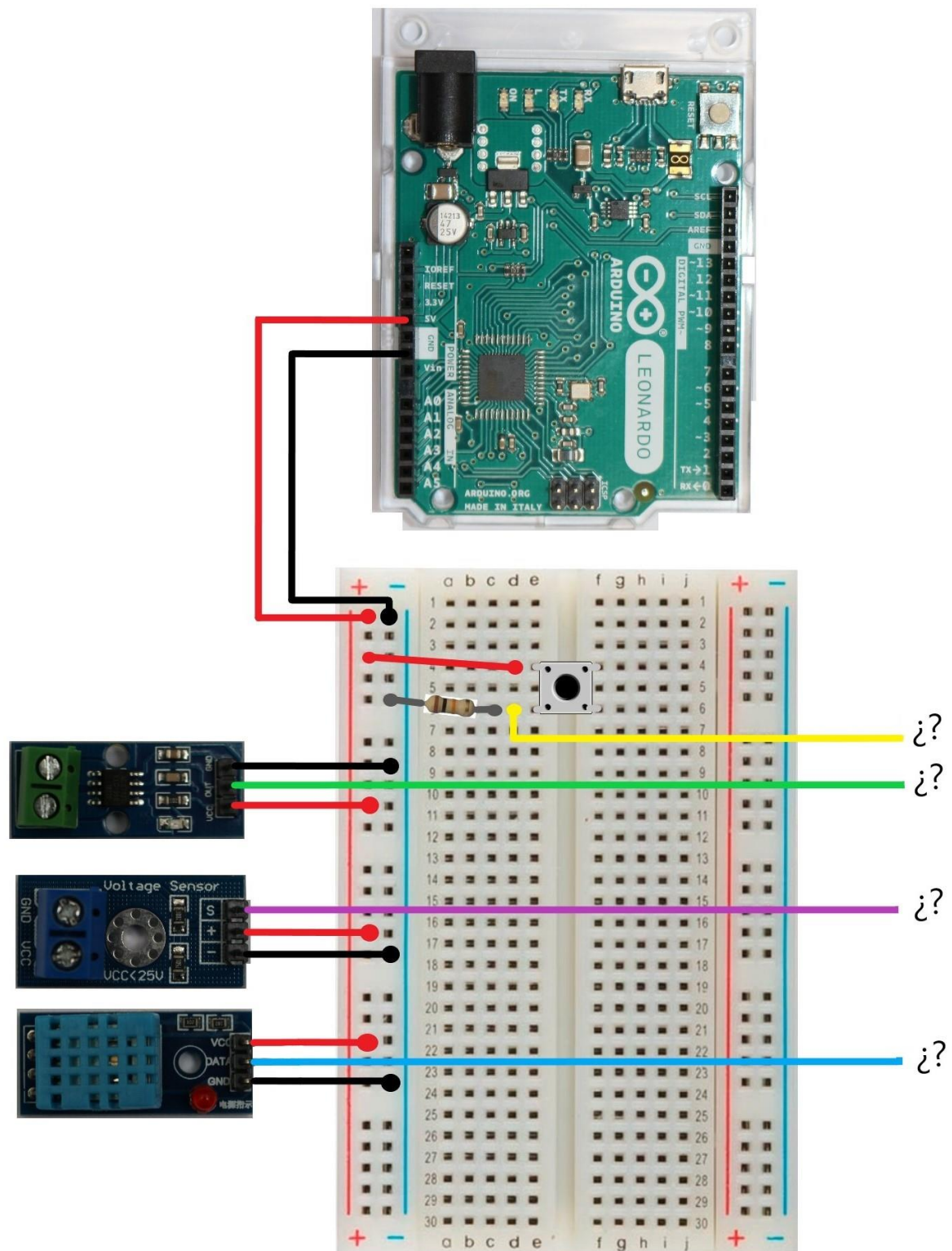


Figura 19. Esquema conexión entre Arduino y los sensores.

Arduino es una herramienta muy sencilla para la interconexión de sensores, pero tendremos que realizar una pequeña configuración como podemos ver en la Figura 19. Para facilitar la conexión en la *protoboard* tenemos una “fila” con la toma de 5V de Arduino y en la “fila adjunta” la toma de tierra de Arduino, así nos será mucho más cómodo conectar los terminales de VCC y GND de los sensores.

Como hemos mencionado anteriormente, tanto el sensor de voltaje como el sensor de corriente devuelven datos en formato analógico, así que **hay que tener mucho cuidado** sobre dónde se conectan estos sensores. Por otro lado, el pulsador y el sensor de temperatura trabajan en formato digital así que **también hay que tener cuidado** en su interconexión. El esquema de la Figura 19 muestra partes sin completar que debéis rellenar.

Podéis intentar **rellenar** el siguiente esquema, en la parte izquierda se encuentran los sensores con el color de cable correspondiente de la Figura 19, a la derecha las entradas que debemos usar en Arduino. Para ello unir cada elemento con su correspondiente.



**Figura 20.** Elección de los pines en Arduino.

Se debe tener en cuenta que los sensores analógicos pueden funcionar con cualquier entrada analógica, pero a la hora de declarar el sensor en el código de Arduino **debemos seguir el mismo orden** como veremos en el siguiente apartado del código de Arduino.

Podéis revisar en cualquier momento la documentación asociada a cada sensor de forma individual en los siguientes apartados:

- Sensor voltaje FZ0430, capítulo 3.1, pag 5.
- Sensor corriente ACS712, capítulo 3.2, pag 7.
- Sensor temperatura DHT11, capítulo 3.3, pag 9.
- Pulsador 4 patas, capítulo 3.4, pag 11.

Anotaciones:



## 5.3 Código Arduino

Una vez se ha terminado de conectar todo el circuito, es hora de interconectar todo mediante Arduino, para comprobar la configuración del IDE de Arduino podéis acudir al apartado 3 que trata sobre este tema. Una vez terminada la conexión entre los sensores y Arduino, es necesario conocer exactamente el pin que hemos usado para indicar este pin dentro del IDE de Arduino. Los cuadros de la Figura 21 tapan el pin que hemos seleccionado anteriormente y se corresponde con los pines de la placa Arduino usados.

Ejemplo de declaración de sensores:

Entrada digital → `const int pinDigital = X`, siendo x el numero de la entrada.

Entrada analógica → `const int pinAnalógico = AX`, siendo x el numero de la entrada.

```

#include "DHT.h"
#define dht11 DHT11
const int pinPulsador = 12; const int pinTemperatura = 18; const int PinA0Voltaje = A0;
int valorPulsador = 0; int sensorVoltajeLectura; float voltaje; float Sensibilidad=0.185;
DHT dht(pinTemperatura, dht11);
void setup() {
  Serial.begin(9600);
  pinMode(pinPulsador, INPUT);
  dht.begin();
}
void loop(){
  valorPulsador = digitalRead(pinPulsador);
  if (valorPulsador == HIGH) {
    sensorVoltajeLectura = analogRead(PinA0Voltaje);
    voltaje = cambiarEscala(sensorVoltajeLectura, 0, 1023, 0.0, 25.0);
    float corrienteSensor= analogRead(A1) * (5.0 / 1023.0);
    float I=((corrienteSensor-2.5)/Sensibilidad)*100;
    float temperatura = dht.readTemperature();
    if(voltaje==0){I=0;}
    Serial.print(temperatura); Serial.print(" °C ");
    Serial.print(I,2); Serial.print(" mA ");
    Serial.print(voltaje);Serial.println("V");
  }
  delay(500);
}
float cambiarEscala(float x, float in_min, float in_max, float out_min, float out_max) {
  return (x - in_min) * (out_max - out_min) / (in_max-in_min)+out_min;
}
  
```

Figura 21. Código Arduino de los sensores.

La declaración de constantes se realiza en la **parte superior del código**, esto sirve para enlazar contantes a nuestros pines de entrada de Arduino facilitando la compresión del código y también declaramos las librerías que necesitemos, en nuestro caso para el sensor de temperatura DHT11 necesitamos su librería específica “DHT.h”.

Para declarar el pin del sensor DHT11 tenemos que hacer uso del método dht(“pinDigital, dht11”) de las librerías que hemos exportado.

```
#include "DHT.h"
#define dht11 DHT11
const int pinPulsador = 2; const int pinTemperatura = 3; const int PinA0Voltaje = 4;
int valorPulsador = 0; int sensorVoltajeLectura; float voltaje; float Sensibilidad=0.185;
DHT dht(pinTemperatura, dht11);
```

**Figura 22.** Declaración de constantes.

La declaración del *void setup()* sirve para configurar algunas cosas básicas como la velocidad de conexión del puerto USB siendo 9600 baudios, para iniciar métodos de las librerías importadas y para declarar los pines digitales usados en la placa.

```
void setup() {
    Serial.begin(9600);
    pinMode(pinPulsador, INPUT);
    dht.begin();
}
```

**Figura 23.** Declaración de void setup().

En la declaración del *void loop()* nos encontramos con un *if (valorPulsador ==HIGH){}* que envuelve todo el código, esto es debido a cuando mantengamos pulsado el pulsador durante más de 500ms se ejecutan las líneas de código que realizan la captura de los datos de los sensores que tenemos conectados a Arduino.

```
void loop(){
    valorPulsador = digitalRead(pinPulsador);
    if (valorPulsador == HIGH) {
        sensorVoltajeLectura = analogRead(PinA0Voltaje);
        voltaje = cambiarEscala(sensorVoltajeLectura, 0, 1023, 0.0, 25.0);
        float corrienteSensor= analogRead(5) * (5.0 / 1023.0);
        float I=((corrienteSensor-2.5)/Sensibilidad)*100;
        float temperatura = dht.readTemperature();
        if(voltaje==0){I=0;}
        Serial.print(temperatura); Serial.print(" °C ");
        Serial.print(I,2); Serial.print(" mA ");
        Serial.print(voltaje);Serial.println("V");
    }
    delay(500);
}
```

**Figura 24.** Declaración de void loop().

## 5.4 Aplicación para captura de datos.

Arduino permite obtener los datos a través del monitor serie, Figura 6. Pero obtener los datos por consola complica extraer los datos para después analizarlos. Por esta razón usaremos una pequeña aplicación que se conecta a través del monitor serie de Arduino para obtener los datos y presentarlos de una mejor manera, así como extraer una hoja Excel al finalizar la comunicación.

1. Iniciamos la aplicación, para ello ejecutar el fichero de la siguiente ruta :

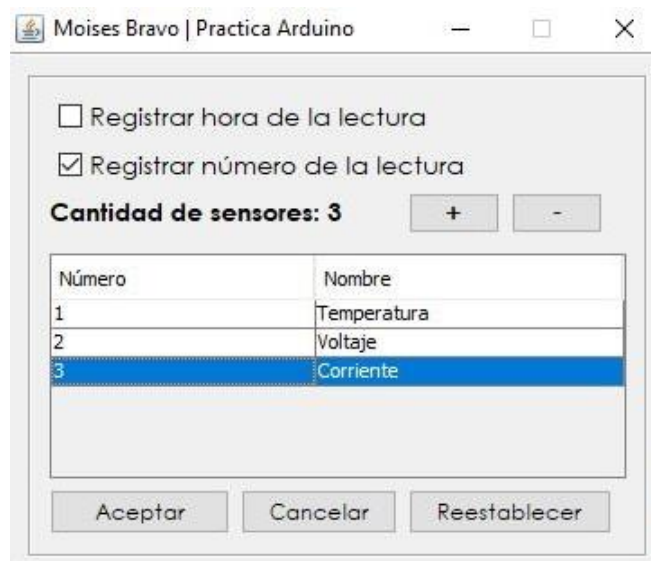
**C:\CarpetaAlmacenada\TFG\ArduinoDataLogger-2.0.0\ArduinoDataLogger-2.0.0\dist**

2. Una vez iniciada veremos la pantalla principal, pulsamos conectar Arduino.



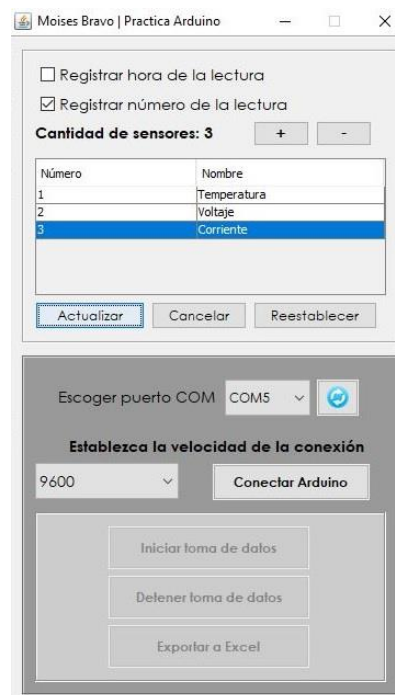
**Figura 25.** Página principal de la aplicación de captura de datos.

3. Añadimos los sensores en el siguiente orden.



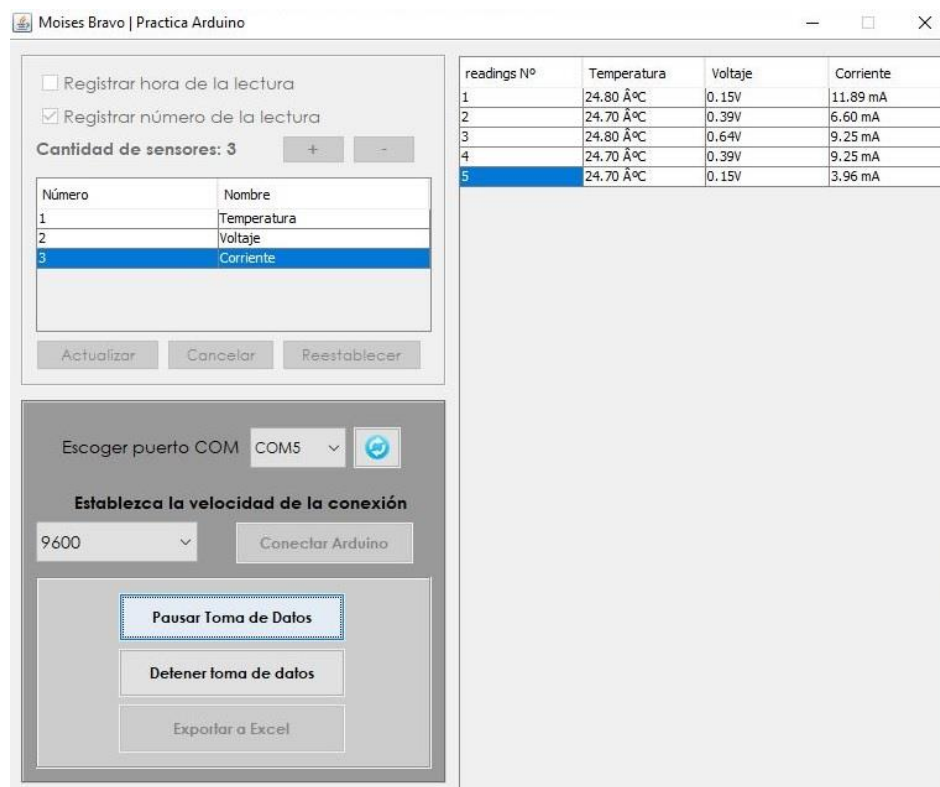
**Figura 26.** Ingreso de los sensores en la aplicación.

4. Pulsamos conectar con Arduino, teniendo en cuenta el puerto COM del USB.



**Figura 27.** Menú para la conexión con Arduino.

5. Una vez que la aplicación esté conectada con Arduino, aquí se mostrarán los datos obtenidos de los sensores al presionar el pulsador. Una vez finalizada la captura de datos se puede exportar el fichero a Excel.



**Figura 28.** Aplicación capturando datos de Arduino.

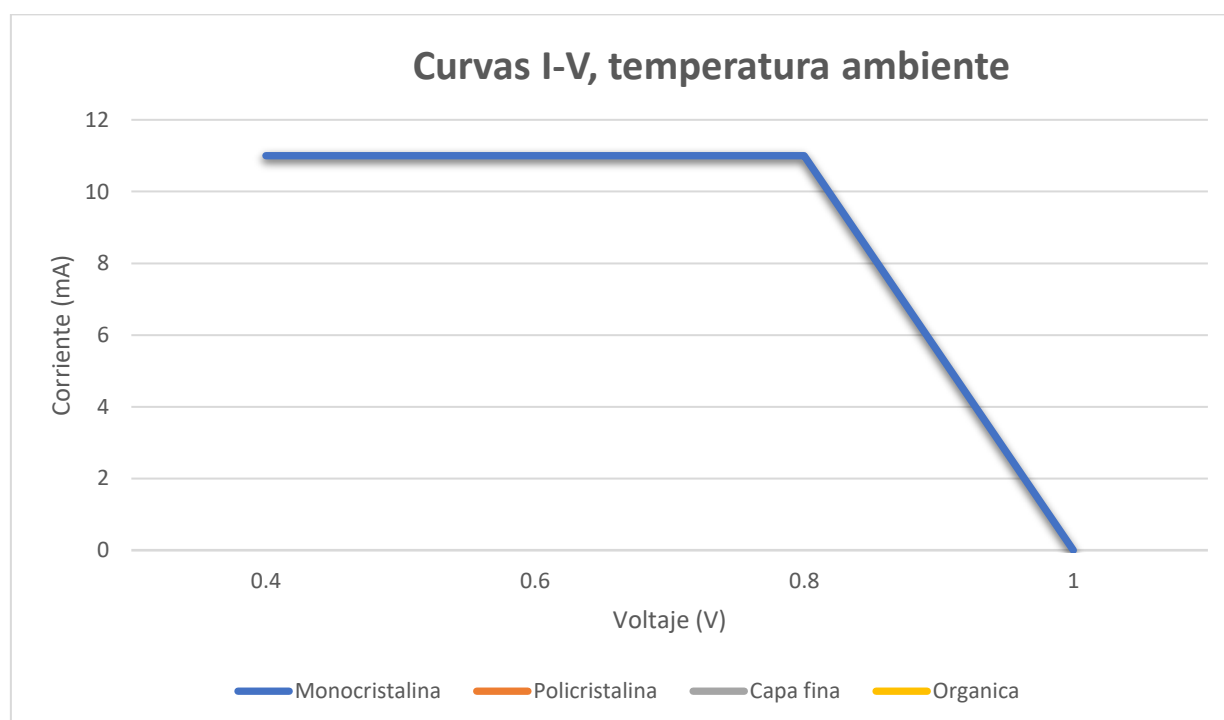
## 5.5 Captura datos a temperatura ambiente.

Una vez que tengamos todo configurado y funcionando es hora de empezar a realizar capturas de los sensores a distintas alturas, completar la siguiente tabla:

	Irradiancia ( $W/m^2$ )	Temperatura ( $^{\circ}C$ )	Corriente (mA)	Voltaje V
<b>Célula monocristalina</b>				
Distancia de 20 cm	4.6	25.6	11.25	0.62
Distancia de 15 cm	7.8	25.6	10.45	0.78
Distancia de 10 cm	14.4	25.2	8.96	0.94
<b>Célula policristalina</b>				
Distancia de 20 cm				
Distancia de 15 cm				
Distancia de 10 cm				
<b>Célula capa fina</b>				
Distancia de 20 cm				
Distancia de 15 cm				
Distancia de 10 cm				
<b>Célula orgánica</b>				
Distancia de 20 cm				
Distancia de 15 cm				
Distancia de 10 cm				

**Figura 29.** Fichero con los datos obtenidos de Arduino a temperatura ambiente.

Después de capturar todos los datos, rellenar las curvas I-V de todas las células:



**Figura 30.** Curva I-V obtenida con los datos de Arduino a temperatura ambiente.

\*Debido a la situación actual del COVID-19, solo he tenido acceso a una célula monocristalina.

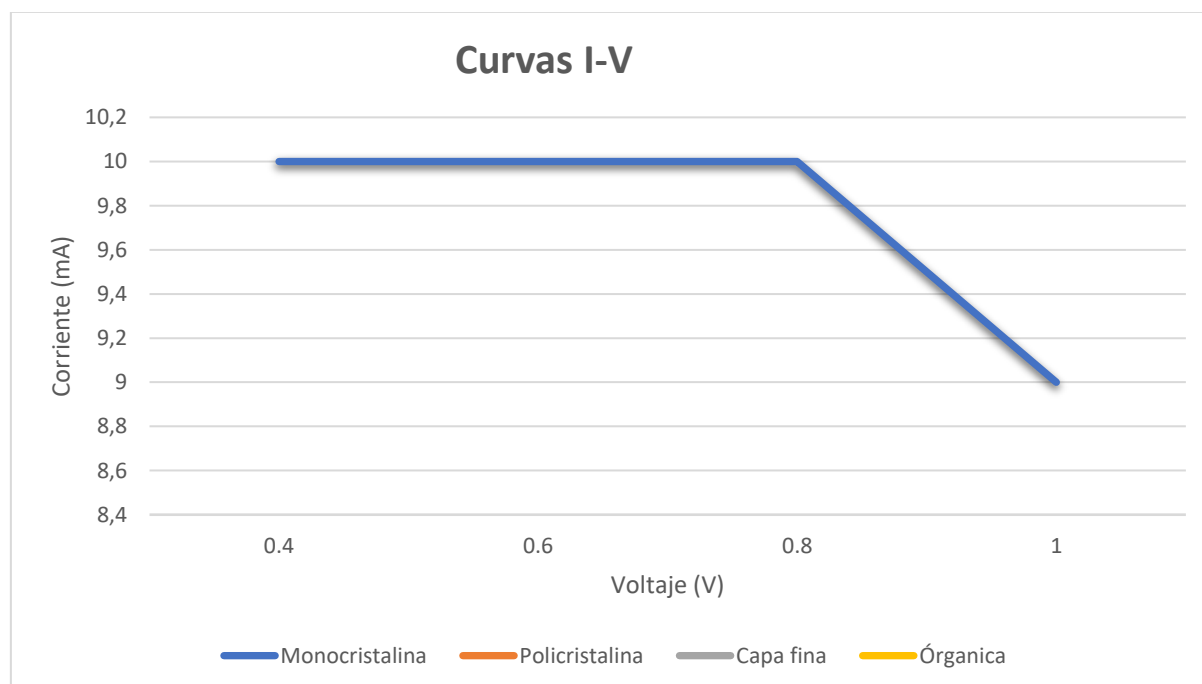
## 5.6 Captura datos a mayor temperatura.

Ahora vamos a repetir las mediciones que hemos realizado, pero aumentando la temperatura ambiente, para ello utilizaremos una pistola estándar de calor industrial consiguiendo subir la temperatura gracias al reducido espacio interno del soporte.

	Irradiancia (W/m <sup>2</sup> )	Temperatura (°C)	Corriente (mA)	Voltaje V
<b>Célula monocristalina</b>				
Distancia de 20 cm	4.7	56.80	11.32	0.54
Distancia de 15 cm	7.4	55.30	10.60	0.69
Distancia de 10 cm	13.6	54.60	9.96	0.82
<b>Célula policristalina</b>				
Distancia de 20 cm				
Distancia de 15 cm				
Distancia de 10 cm				
<b>Célula capa fina</b>				
Distancia de 20 cm				
Distancia de 15 cm				
Distancia de 10 cm				
<b>Célula orgánica</b>				
Distancia de 20 cm				
Distancia de 15 cm				
Distancia de 10 cm				

**Figura 31.** Fichero con los datos obtenidos de Arduino con temperatura alta.

Después de capturar todos los datos, rellenar las curvas I-V de todas las células:



**Figura 32.** Curva I-V obtenida con los datos de Arduino a temperatura alta.

\*Debido a la situación actual del COVID-19, solo tenemos acceso a una célula monocristalina.



## 7. Presupuesto

Fecha presupuesto	17/06/2020	Validez:	2020
DESCRIPCIÓN	UNIDADES	PRECIO	TOTAL
Placa Arduino Leonardo	1,00	22,00	€22,00
Sensor FZ0430 Voltaje	1,00	1,45	€1,45
Sensor ACS712 Corriente	1,00	2,96	€2,96
Sensor DHT11 Temperatura	1,00	1,99	€1,99
Pulsador 4 patas	1,00	1,89	€1,89
Cables interconexión	1,00	8,89	€8,89
Soporte impresora 3D	0,63	25,00	€15,75
Tiras Led	2,00	8,00	€16,00
Fuente alimentación Led	1,00	9,00	€9,00
Protoboard	1,00	6,50	€6,50
TOTAL			€86,43

## 8. Conclusiones y líneas futuras

Uno de los principales objetivos de esta práctica para la asignatura “Energías y Telecomunicaciones ” es que el alumno tenga un conocimiento básico sobre las tecnologías existentes de células fotovoltaicas en el mercado y a su vez tenga claro las ventajas y desventajas de usar una u otra tecnología. En la practica analizamos los cuatro tipos de células más usadas durante años cumplimento este objetivo, además en caso de que surja alguna nueva tecnología sería muy sencillo añadirla en la práctica, por ese motivo esta práctica es fácilmente actualizable durante los próximos años.

El segundo objetivo de la práctica es que el alumno se introduzca y conozca el mundo del hardware open-source porque cada vez es más frecuente ver este tipo de proyectos en el entorno profesional, el alumno al finalizar la practica tendrá una pequeña noción de esta tecnología, cumpliendo con nuestro objetivo también.

Para un futuro proyecto que mejore esta práctica se propone diseñar una aplicación similar a la usada en este proyecto para capturar los datos de Arduino, pero que permita generar la curva I-V instantáneamente después de la captura de los datos. En cuanto a esta práctica este es el punto débil, ya que el alumno debe representar la grafica en el guion de la práctica. Por último, en nuestro proyecto usaremos un pistola de calor industrial, se propone también buscar una forma mas eficiente de aumentar la temperatura de la célula.